



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de La Paz



Últimos Avances en Robótica y Computación



**TECNOLÓGICO
NACIONAL DE MÉXICO**

Coordinadores de la Edición
Iliana Castro Liera
Mario Cortés Larrinaga



ISBN: 978-607-98174-9-7



9 786079 817497

Esta página fue dejada en blanco intencionalmente.



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de La Paz

Últimos Avances en Robótica y Computación

D. R. ©INSTITUTO TECNOLÓGICO DE LA PAZ
Blvd. Forjadores de B.C.S. No. 4720
Col. 8 de octubre
C.P. 23080
La Paz Baja California Sur, México

Primera edición agosto de 2021
ISBN: 978-607-98174-9-7



Reservados todos los derechos. Ninguna parte de este libro puede ser reproducida, archivada o transmitida, en cualquier sistema –electrónico, mecánico, de fotorreproducción, de almacenamiento en memoria o cualquier otro– sin hacerse acreedor a las sanciones establecidas en las leyes, salvo con el permiso escrito del titular de los derechos de la obra.

Diseño de la portada: Iliana Castro Liera

Esta página fue dejada en blanco intencionalmente.

Coordinador General

M.S.C. Iliana Castro Liera- Tecnológico Nacional de México / I. T. de La Paz

Presidente del Programa Técnico

Dr. Miguel Ángel Llama Leal - Tecnológico Nacional de México / I. T. de La Laguna

Comité Revisor

Dr. Carlos Álvarez Macías - Tecnológico Nacional de México / I. T. de La Laguna

Dr. Ramón García Hernández - Tecnológico Nacional de México / I. T. de La Laguna

Dr. Alejandro Enrique Dzul López - Tecnológico Nacional de México / I. T. de La Laguna

Dr. Miguel Ángel Llama Leal - Tecnológico Nacional de México / I. T. de La Laguna

Dr. Juan Sifuentes Mijares - Tecnológico Nacional de México / I. T. de La Laguna

Dr. Juan Felipe Guerra Cano - Tecnológico Nacional de México / I. T. de La Laguna

Dr. Rodrigo Loera Palomo - Cátedras CONACyT /Tecnológico Nacional de México / I. T. de La Laguna

Dr. José Armando Sáenz Esqueda - Universidad Juárez del Estado de Durango

Dr. Jaime González Siserra - Universidad Politécnica de Pachuca

Dr. Jesús Alberto Sandoval Galarza - Tecnológico Nacional de México / I. T. de La Paz

Dr. Saúl Martínez Díaz - Tecnológico Nacional de México / I. T. de La Paz

Dr. Israel Marcos Santillán Méndez - Tecnológico Nacional de México / I. T. de La Paz

Dr. Marco Antonio Castro Liera - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. Iliana Castro Liera - Tecnológico Nacional de México / I. T. de La Paz

M.C. Jorge Enrique Luna Taylor - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. José Luis Gómez Torres - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. César Higuera Verdugo - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. Javier Alberto Carmona Troyo - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. David Higuera Balderrama - Tecnológico Nacional de México / I. T. de La Paz

M.A.T.I. Luis Armando Cárdenas Florido- Tecnológico Nacional de México / I. T. de La Paz

M.S.C. Joel Artemio Morales Viscaya - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. Jacob Ezequiel Merecías Pérez - Tecnológico Nacional de México / I. T. de La Paz

M.S.C. José Luis Gómez Torres - Tecnológico Nacional de México / I. T. de La Paz

M.C. Germán Meléndrez Carballo - CICESE

M.C. Karen Guerrero Vázquez - CICESE

Coordinadores de la edición

M.S.C. Iliana Castro Liera - Tecnológico Nacional de México / I. T. de La Paz

Dr. Mario Cortés Larrinaga - Tecnológico Nacional de México / I. T. de La Paz

Índice general

Portada	I
Comités	V
Índice	VI
Control Basado en RedesNeuronales Adaptables para el Seguimiento de Trayecto- rias de un Quadrotor	1
Experimentación y simulación de un sistema accesible para implementar algoritmos de control clásicos y discontinuos en motores DC	8
Sintonía con PSO de un controlador PID para la regulación de posición de una articulación motriz pendular	14
Estado actual de los coches eléctricos en México, 2021	24
Máquina de Vectores de Soporte para la Detección de Datos Anómalos en Dispo- sitivos IoT	31
Realización en FPGA de Sistema Encriptación de Imágenes con Atractores Caóti- cos: Dos-Alas y Jerk	37
Implementación en Tarjeta de FPGA de Dos Sistemas Caóticos Conservativos Ha- miltonianos	46
Detección de Microaneurismas en Retinografía Mediante el Uso de Operaciones Morfológicas	53
Sistema CAD para la Detección de Nódulos Pulmonares Basado en Características Estadísticas y de Textura	59
Detección e identificación de manchas en paneles solares con técnicas de visión artificial	65
Sistema de percepción no centralizado para enjambres de robots RAOI basado en Deep Learning	73
Sistema de Navegación de Bajo Costo Basado en Visión Artificial	80
Detección de automóviles en un estacionamiento haciendo uso de las redes neuro- nales pulsantes	87
License plate detection based on Intersecting Cortical Model and entropy signatures	96

Control Basado en Redes Neuronales Adaptables para el Seguimiento de Trayectorias de un Quadrotor*

Ivan Lopez-Sanchez
 Instituto Politécnico Nacional-CITEDI
 Tijuana, B.C., México
 ilopez@citedi.mx

Ricardo Pérez-Alcocer
 CONACYT-Instituto Politécnico
 Nacional-CITEDI
 Tijuana, B.C., México
 rrperez@citedi.mx

Javier Moreno-Valenzuela
 Instituto Politécnico Nacional-CITEDI
 Tijuana, B.C., México
 moreno@citedi.mx

Resumen—En este documento se presenta el desarrollo de un controlador de seguimiento de trayectorias basado en redes neuronales adaptables usando tangente hiperbólica como función de activación. El desarrollo del controlador se hace bajo la suposición de la existencia de un controlador interno al que no se tiene acceso, de forma que el controlador propuesto funge como un lazo de control externo que genera señales de referencia para el controlador interno. El controlador propuesto está diseñado de tal forma que no requiere del conocimiento de los parámetros del quadrotor ni los del controlador interno. Se presenta el análisis de convergencia del error de seguimiento. Además, el controlador propuesto se compara en simulación con respecto a un esquema de control tipo proporcional-integral-derivativo (PID) obteniendo mejor desempeño con el control propuesto.

Palabras clave—Control inteligente, redes neuronales adaptables, quadrotor, control de doble lazo.

I. INTRODUCCIÓN

En la actualidad, los vehículos aéreos no tripulados (UAVs por sus siglas en inglés) han captado la atención de la comunidad científica e industrial. De entre los diversos tipos de UAVs los quadrotores han destacado debido a su relativa simplicidad, su capacidad de realizar despegues y aterrizajes verticales, su habilidad de permanecer suspendidos en el aire y su tamaño compacto. Si bien estas aeronaves fueron concebidas con fines militares, su gran versatilidad las ha convertido en herramientas útiles en diversos campos de aplicación, por ejemplo, agricultura [1], control de incendios forestales [2], protección de vida silvestre [3], ubicación de personas en zonas de desastre [4], entre muchas otras.

Debido a la variedad de aplicaciones para dichas aeronaves, surge la necesidad de desarrollar controladores que garanticen el cumplimiento de sus respectivas misiones. A su vez, cada campo de aplicación tiene sus propios desafíos y complicaciones, por esto, es que diversos esquemas de control se pueden encontrar en la literatura. Distintas propuestas se han hecho con el objetivo de estabilizar y controlar un quadrotor, algunas basadas en técnicas de control lineal, como control PID [5], control LQR [6] y control H_∞ [7]. De igual forma se han empleado técnicas de control no lineal, como esquemas por modos deslizantes [8], linealización

por retroalimentación [9], backstepping [10], control por predicción del modelo [11], control adaptable [12] y control usando la técnica de saturación anidada [13]. Además, un enfoque que ha resultado atractivo es el uso de esquemas de control inteligente como son los de lógica difusa [14], aprendizaje por refuerzo [15], aprendizaje iterativo [16] y redes neuronales ya sean entrenadas [17] o adaptables [18].

De entre todas estas propuestas de control, aquellas basadas en redes neuronales adaptables han demostrado ser efectivas para el rechazo de perturbaciones y compensación de la dinámica no modelada [19]–[21]. Calidad que resulta útil en el control de quadrotores, ya que en su entorno de trabajo están expuestos a diversos fenómenos ambientales que perturban al sistema. Además, en algunas aplicaciones como el transporte de cargas suspendidas por cable, la dinámica de la carga influye en el comportamiento de la aeronave, efecto que también puede ser mitigado de forma efectiva a través de la implementación de redes neuronales [22], [23].

En este trabajo se propone un esquema de control basado en redes neuronales adaptables. El esquema propuesto es desarrollado y aplicado a un quadrotor que ya cuenta con un controlador interno, lo que conduce a un sistema que tiene como entrada señales normalizadas de pose y velocidad y tiene como salida la posición y el ángulo de *yaw*. La red neuronal es una parte importante del esquema propuesto ya que permite el rechazo de perturbaciones y compensación de la dinámica no modelada, lo que es una cualidad importante de esta propuesta ya que no se requiere del conocimiento de los parámetros de la aeronave. La ley de adaptación con la que los pesos de salida de la red neuronal son estimados es obtenida a través del análisis de convergencia del error, con el cual se garantiza el seguimiento de la trayectoria deseada y el acotamiento del error de estimación de los pesos de la red neuronal. La propuesta es respaldada por una comparación en simulación donde se prueba la funcionalidad del esquema y la utilidad de la red neuronal con respecto a un esquema de control tipo PID.

El resto del documento está organizado de la siguiente manera. En la sección II se presenta el sistema que resulta del modelo dinámico de un quadrotor y la presencia de un controlador interno. El esquema de control propuesto y la red neuronal se describen en la sección III. El análisis

*Trabajo apoyado por el Consejo Nacional de Ciencia y Tecnología (Proyecto Apoyado por el Fondo Sectorial de Investigación para la Educación), México, bajo el proyecto A1-S-24762.

de convergencia del error del sistema en lazo cerrado se describe en la sección IV. Los resultados de la comparación en simulación se presentan en la sección V. Por último, las conclusiones de este trabajo se establecen en la sección VI.

Notación: La notación utilizada en este documento se describe a continuación. $\|\mathbf{x}\|_1$ representa la norma l_1 del vector $\mathbf{x} \in \mathbb{R}^n$ definida como $\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$. $\|\mathbf{x}\| = \sqrt{\mathbf{x}^T \mathbf{x}}$ representa la norma euclidiana del vector $\mathbf{x} \in \mathbb{R}^n$. $\lambda_{\min}\{A\}$ y $\lambda_{\max}\{A\}$ representan el valor propio mínimo y máximo de la matriz simétrica y positiva definida $A \in \mathbb{R}^{n \times n}$. $\text{Tr}\{A\}$ representa la traza de la matriz $A \in \mathbb{R}^{n \times n}$ definida como $\text{Tr}\{A\} = \sum_{i=1}^n A_{ii}$.

II. MODELO MATEMÁTICO DE UN QUADROTOR Y OBJETIVO DE CONTROL

Según lo discutido en [24]–[27], muchos de los quadrotors comerciales poseen un controlador interno que se encarga de estabilizar el vehículo y sirve como intérprete entre los comandos dados por el usuario y las entradas de control propias del modelo dinámico del quadrotor. Por lo tanto, el sistema que representa la dinámica del quadrotor sujeto al controlador interno se puede representar de la siguiente manera

$$\ddot{\mathbf{x}}^w = T(\psi)K_u \mathbf{u} - K_v \dot{\mathbf{x}}^w, \quad (1)$$

donde $\mathbf{x}^w = [x \ y \ z \ \psi]^T \in \mathbb{R}^4$ es un vector que contiene la posición en el espacio tridimensional (x, y, z) y el ángulo de yaw ψ con respecto del marco de referencia inercial, $K_u \in \mathbb{R}^{4 \times 4}$ y $K_v \in \mathbb{R}^{4 \times 4}$ son matrices diagonales y definidas positivas que contienen los parámetros del vehículo y del controlador interno, descritas explícitamente como

$$K_u = \begin{bmatrix} g\theta_{\max} & 0 & 0 & 0 \\ 0 & g\phi_{\max} & 0 & 0 \\ 0 & 0 & \frac{\dot{z}_{\max}}{\tau_z} & 0 \\ 0 & 0 & 0 & \frac{\dot{\psi}_{\max}}{\tau_{\psi}} \end{bmatrix},$$

$$K_v = \begin{bmatrix} d_x & 0 & 0 & 0 \\ 0 & d_y & 0 & 0 \\ 0 & 0 & \left(\frac{1}{\tau_z} + \frac{d_z}{m}\right) & 0 \\ 0 & 0 & 0 & \frac{1}{\tau_{\psi}} \end{bmatrix},$$

donde g es la constante de aceleración gravitacional, m es la masa del quadrotor, θ_{\max} y ϕ_{\max} representan los ángulos de *roll* y *pitch* máximos, respectivamente, \dot{z}_{\max} es la velocidad vertical máxima, $\dot{\psi}_{\max}$ es la velocidad angular máxima del quadrotor sobre su propio eje vertical, τ_z y τ_{ψ} son constantes relacionadas a los tiempos de respuesta del sistema y d_x , d_y y d_z son los coeficientes de arrastre aerodinámico correspondientes a los ejes x , y y z , respectivamente. La matriz $T(\psi) \in \mathbb{R}^{4 \times 4}$ es una matriz de transformación definida como

$$T(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 & 0 \\ \sin(\psi) & -\cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

y $\mathbf{u} = [u_\theta \ u_\phi \ u_z \ u_{\dot{\psi}}]^T \in \mathbb{R}^4$ es el vector de entradas de control cuyo valor está normalizado y es adimensional, donde:

- u_θ es la entrada de posición angular relacionada al desplazamiento en el eje x del marco de referencia inercial.
- u_ϕ es la entrada de posición angular relacionada al desplazamiento en el eje y del marco de referencia inercial.
- u_z es la entrada de velocidad relacionada al movimiento en el eje z del marco de referencia inercial.
- $u_{\dot{\psi}}$ es la entrada de velocidad angular relacionada a la rotación sobre el eje z en el marco de referencia inercial.

El sistema en (1) describe el comportamiento del quadrotor en condiciones ideales. Sin embargo, durante aplicaciones en exteriores, los parámetros del modelo pueden cambiar como consecuencia de las condiciones climáticas (ráfagas de viento, lluvia, cambios en la densidad del aire, etc.), o también, según la tarea a desempeñar, como por ejemplo, transportar una carga suspendida. Por consiguiente, el modelo en (1) puede ser expresado como

$$\ddot{\mathbf{x}}^w = T(\psi)K_u \mathbf{u} - K_v \dot{\mathbf{x}}^w + \boldsymbol{\delta}(t), \quad (2)$$

donde $\boldsymbol{\delta}(t) \in \mathbb{R}^4$ representa el vector de perturbaciones externas y de dinámica no modelada, el cual se asume acotado como sigue

$$\|\boldsymbol{\delta}(t)\| \leq \delta_0,$$

donde δ_0 es una constante positiva.

Objetivo de control

Definiendo el error de seguimiento de trayectoria como

$$\mathbf{e} = \mathbf{x}_d^w - \mathbf{x}^w \in \mathbb{R}^4, \quad (3)$$

donde $\mathbf{x}_d^w(t) = [x_d(t) \ y_d(t) \ z_d(t) \ \psi_d(t)]^T \in \mathbb{R}^4$ es el vector de señales deseadas, el objetivo es diseñar un controlador $\mathbf{u}(t) = [u_\theta(t) \ u_\phi(t) \ u_z(t) \ u_{\dot{\psi}}(t)]^T \in \mathbb{R}^4$ tal que

$$\lim_{t \rightarrow \infty} \mathbf{e}(t) = 0$$

sea garantizado.

III. CONTROLADOR BASADO EN REDES NEURONALES ADAPTABLES

El controlador propuesto está basado en una red neuronal adaptable usando tangente hiperbólica como función de activación. Cabe resaltar que dicha propuesta fue desarrollada con el objetivo de prescindir del conocimiento de los parámetros tanto de la dinámica del vehículo como del controlador interno.

Con base en la definición del error en (3) se obtiene la dinámica del error dada por

$$\ddot{\mathbf{e}} = \ddot{\mathbf{x}}_d^w - T(\psi)K_u \mathbf{u} + K_v \dot{\mathbf{x}}^w - \boldsymbol{\delta}(t). \quad (4)$$

Considerando la estructura de la matriz K_u y bajo la suposición de que $\theta_{\max} = \phi_{\max}$, se puede multiplicar K_u^{-1} en ambos miembros de (4) obteniendo así

$$K_u^{-1} \ddot{\mathbf{e}} = K_u^{-1} [\ddot{\mathbf{x}}_d^w + K_v \dot{\mathbf{x}}^w - \boldsymbol{\delta}(t)] - T(\psi) \mathbf{u},$$

expresión que puede reescribirse como

$$\Lambda \ddot{e} = \mathbf{f}(\ddot{\mathbf{x}}_d^w, \dot{\mathbf{x}}^w) - T(\psi)\mathbf{u}, \quad (5)$$

donde $\Lambda = K_u^{-1}$ y $\mathbf{f}(\ddot{\mathbf{x}}_d^w, \dot{\mathbf{x}}^w) = \Lambda[\ddot{\mathbf{x}}_d^w + K_v \dot{\mathbf{x}}^w - \delta(t)]$.

Haciendo uso de la propiedad de aproximación universal de las redes neuronales, la función $\mathbf{f}(\ddot{\mathbf{x}}_d^w, \dot{\mathbf{x}}^w)$ puede ser estimada por una red neuronal como

$$\mathbf{f}(\ddot{\mathbf{x}}_d^w, \dot{\mathbf{x}}^w) = W^T \boldsymbol{\sigma}(V^T \boldsymbol{\gamma}) + \boldsymbol{\epsilon}, \quad (6)$$

donde $\boldsymbol{\gamma} = [1 \ \ddot{\mathbf{x}}_d^w \ \dot{\mathbf{x}}^w]^T \in \mathbb{R}^9$, es el vector extendido de entradas de la red neuronal, $V \in \mathbb{R}^{9 \times L_N}$ y $W \in \mathbb{R}^{L_N \times 4}$ son las matrices de pesos de entrada y salida ideales, respectivamente, siendo $\mathbf{p} = V^T \boldsymbol{\gamma}$, $\boldsymbol{\sigma}(\mathbf{p}) \in \mathbb{R}^{L_N}$ es el vector de funciones de activación definido como $\boldsymbol{\sigma}(\mathbf{p}) = [\tanh(p_1) \ \cdots \ \tanh(p_{L_N})]^T$, $\boldsymbol{\epsilon} \in \mathbb{R}^4$ es el error de aproximación de la red neuronal y finalmente, $L_N \in \mathbb{N}$ es la cantidad de neuronas que componen a la red neuronal.

Con base en las ecuaciones (5) y (6) se propone el siguiente controlador neuronal

$$\begin{aligned} \mathbf{u} = & T(\psi)^{-1} \left[K_p \mathbf{e} + K_d \dot{\mathbf{e}} + \hat{W}^T \boldsymbol{\sigma}(V^T \boldsymbol{\gamma}) \right. \\ & \left. + \Delta \text{sign}(\alpha \mathbf{e} + \dot{\mathbf{e}}) \right], \end{aligned} \quad (7)$$

con la ley de adaptación de pesos dada por

$$\dot{\hat{W}} = N \boldsymbol{\sigma}(\alpha \mathbf{e} + \dot{\mathbf{e}})^T - N \hat{W} k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\|, \quad (8)$$

donde $N \in \mathbb{R}^{L_N \times L_N}$, $K_p \in \mathbb{R}^{4 \times 4}$, $K_d \in \mathbb{R}^{4 \times 4}$ y $\Delta \in \mathbb{R}^{4 \times 4}$ son matrices diagonales definidas positivas y $\alpha, k \in \mathbb{R}$ son constantes positivas. Además, para un vector general $\mathbf{x} \in \mathbb{R}^4$, el vector de funciones signo está dado por $\text{sign}(\mathbf{x}) = [\text{sign}(x_1) \ \cdots \ \text{sign}(x_4)]^T$, donde

$$\text{sign}(x_i) = \begin{cases} 1 & \text{para } x_i > 0, \\ 0 & \text{para } x_i = 0, \\ -1 & \text{para } x_i < 0. \end{cases}$$

IV. ANÁLISIS DE CONVERGENCIA DEL ERROR

Sustituyendo el controlador neuronal (7) en la dinámica del error en (5) se obtiene la dinámica de lazo cerrado dada como

$$\begin{aligned} \Lambda \ddot{e} = & W^T \boldsymbol{\sigma}(V^T \boldsymbol{\gamma}) - \hat{W}^T \boldsymbol{\sigma}(V^T \boldsymbol{\gamma}) - K_p \mathbf{e} - K_d \dot{\mathbf{e}} \\ & - \Delta \text{sign}(\alpha \mathbf{e} + \dot{\mathbf{e}}) + \boldsymbol{\epsilon}, \end{aligned} \quad (9)$$

para simplificar la notación, de aquí en adelante $\boldsymbol{\sigma} = \boldsymbol{\sigma}(V^T \boldsymbol{\gamma})$.

Definiendo el error de estimación de los pesos de salida como

$$\tilde{W} = W - \hat{W} \quad (10)$$

la ecuación (9) puede reescribirse como

$$\Lambda \ddot{e} = \tilde{W}^T \boldsymbol{\sigma} - K_p \mathbf{e} - K_d \dot{\mathbf{e}} - \Delta \text{sign}(\alpha \mathbf{e} + \dot{\mathbf{e}}) + \boldsymbol{\epsilon}. \quad (11)$$

Se propone la siguiente función

$$L = \frac{1}{2} \mathbf{r}^T P \mathbf{r} + \frac{1}{2} \text{Tr} \left\{ \tilde{W}^T N^{-1} \tilde{W} \right\}, \quad (12)$$

donde $\mathbf{r} = [\mathbf{e}^T \ \dot{\mathbf{e}}^T]^T \in \mathbb{R}^8$ y

$$P = \begin{bmatrix} K_p & \alpha \Lambda \\ \alpha \Lambda & \Lambda \end{bmatrix} \in \mathbb{R}^{8 \times 8}.$$

Ahora, para garantizar que (12) es una función positiva definida, se busca una cota inferior. Haciendo uso del hecho que $\mathbf{x}^T A \mathbf{x} \geq \lambda_{\min} \{A\} \|\mathbf{x}\|^2$ y que $\mathbf{x}^T A \mathbf{y} \geq -\lambda_{\max} \{A\} \|\mathbf{x}\| \|\mathbf{y}\| \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, se obtiene la siguiente cota inferior

$$\begin{aligned} L \geq & \frac{1}{2} \lambda_{\min} \{K_p\} \|\mathbf{e}\|^2 - \alpha \lambda_{\max} \{\Lambda\} \|\mathbf{e}\| \|\dot{\mathbf{e}}\| \\ & + \frac{1}{2} \lambda_{\min} \{\Lambda\} \|\dot{\mathbf{e}}\|^2 + \frac{1}{2} \text{Tr} \left\{ \tilde{W}^T N^{-1} \tilde{W} \right\}, \end{aligned}$$

que puede reescribirse como

$$L \geq \frac{1}{2} \mathbf{s}^T \Phi \mathbf{s} + \frac{1}{2} \text{Tr} \left\{ \tilde{W}^T N^{-1} \tilde{W} \right\},$$

donde $\mathbf{s} = [\|\mathbf{e}\| \ \|\dot{\mathbf{e}}\|]^T \in \mathbb{R}^2$ y

$$\Phi = \begin{bmatrix} \lambda_{\min} \{K_p\} & -\frac{1}{2} \alpha \lambda_{\max} \{\Lambda\} \\ -\frac{1}{2} \alpha \lambda_{\max} \{\Lambda\} & \lambda_{\min} \{\Lambda\} \end{bmatrix} \in \mathbb{R}^{2 \times 2},$$

de forma que, para que la matriz Φ sea definida positiva, las siguientes condiciones se deben cumplir

$$\begin{aligned} \lambda_{\min} \{K_p\} & > 0, \\ \lambda_{\min} \{\Lambda\} & > 0, \\ 0 < \alpha < & \frac{2\sqrt{\lambda_{\min} \{K_p\} \lambda_{\min} \{\Lambda\}}}{\lambda_{\max} \{\Lambda\}}. \end{aligned} \quad (13)$$

La derivada temporal de (12) está dada como

$$\begin{aligned} \dot{L} = & \mathbf{e}^T K_p \dot{\mathbf{e}} + (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \Lambda \dot{\mathbf{e}} + \dot{\mathbf{e}}^T \alpha \Lambda \dot{\mathbf{e}} \\ & - \text{Tr} \left\{ \tilde{W}^T N^{-1} \dot{\tilde{W}} \right\}, \end{aligned} \quad (14)$$

con base en la definición (10), se tiene que $\dot{\tilde{W}} = -\dot{\hat{W}}$. Sustituyendo (8) y (11) en (14) se obtiene

$$\begin{aligned} \dot{L} = & -\mathbf{e}^T \alpha K_p \mathbf{e} - \mathbf{e}^T \alpha K_d \dot{\mathbf{e}} - \dot{\mathbf{e}}^T K_d \dot{\mathbf{e}} + \dot{\mathbf{e}}^T \alpha \Lambda \dot{\mathbf{e}} \\ & + (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \tilde{W}^T \boldsymbol{\sigma} - (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \Delta \text{sign}(\alpha \mathbf{e} + \dot{\mathbf{e}}) \\ & + \text{Tr} \left\{ -\tilde{W}^T \boldsymbol{\sigma}(\alpha \mathbf{e} + \dot{\mathbf{e}})^T + \tilde{W}^T \hat{W} k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \right\} \\ & + (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \boldsymbol{\epsilon}. \end{aligned} \quad (15)$$

Usando las propiedades

$$\begin{aligned} \text{Tr} \{A + B\} & = \text{Tr} \{A\} + \text{Tr} \{B\}, \\ \text{Tr} \{cA\} & = c \text{Tr} \{A\}, \forall c \in \mathbb{R}, \end{aligned}$$

los hechos

$$\begin{aligned} (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \tilde{W}^T \boldsymbol{\sigma} & = \text{Tr} \left\{ \tilde{W}^T \boldsymbol{\sigma}(\alpha \mathbf{e} + \dot{\mathbf{e}})^T \right\}, \\ (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \Delta \text{sign}(\alpha \mathbf{e} + \dot{\mathbf{e}}) & = \|\Delta(\alpha \mathbf{e} + \dot{\mathbf{e}})\|_1, \end{aligned}$$

y la definición del error de estimación en (10), la ecuación (15) puede reescribirse como

$$\begin{aligned} \dot{L} = & -\mathbf{e}^T \alpha K_p \mathbf{e} - \mathbf{e}^T \alpha K_d \dot{\mathbf{e}} - \dot{\mathbf{e}}^T K_d \dot{\mathbf{e}} + \dot{\mathbf{e}}^T \alpha \Lambda \dot{\mathbf{e}} \\ & - \|\Delta(\alpha \mathbf{e} + \dot{\mathbf{e}})\|_1 + k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \text{Tr} \left\{ \tilde{W}^T (W - \hat{W}) \right\} \\ & + (\alpha \mathbf{e} + \dot{\mathbf{e}})^T \boldsymbol{\epsilon}, \end{aligned}$$

y usando el hecho que $-\mathbf{x}^T A \mathbf{x} \leq -\lambda_{\min}\{A\} \|\mathbf{x}\|^2$, $\mathbf{x}^T A \mathbf{x} \leq \lambda_{\max}\{A\} \|\mathbf{x}\|^2$ y que $-\mathbf{x}^T A \mathbf{y} \leq \lambda_{\max}\{A\} \|\mathbf{x}\| \|\mathbf{y}\| \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ se obtiene la siguiente expresión

$$\begin{aligned} \dot{L} \leq & -\alpha \lambda_{\min}\{K_p\} \|\mathbf{e}\|^2 + \alpha \lambda_{\max}\{K_d\} \|\mathbf{e}\| \|\dot{\mathbf{e}}\| \\ & -(\lambda_{\min}\{K_d\} - \alpha \lambda_{\max}\{\Lambda\}) \|\dot{\mathbf{e}}\|^2 \\ & -\lambda_{\min}\{\Delta\} \|\alpha \mathbf{e} + \dot{\mathbf{e}}\|_1 \\ & + k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \text{Tr} \left\{ \tilde{W}^T (W - \tilde{W}) \right\} \\ & + \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \|\boldsymbol{\epsilon}\|_1. \end{aligned}$$

Ahora, usando la norma de Frobenius [28] se sabe que

$$\begin{aligned} \text{Tr} \left\{ \tilde{W}^T (W - \tilde{W}) \right\}_F &= \left\langle \tilde{W}, W \right\rangle_F - \|\tilde{W}\|_F^2 \\ &\leq \|\tilde{W}\|_F \|W\|_F - \|\tilde{W}\|_F^2. \end{aligned}$$

Entonces, una cota superior para \dot{L} está dada como

$$\begin{aligned} \dot{L} \leq & -\alpha \lambda_{\min}\{K_p\} \|\mathbf{e}\|^2 + \alpha \lambda_{\max}\{K_d\} \|\mathbf{e}\| \|\dot{\mathbf{e}}\| \\ & -(\lambda_{\min}\{K_d\} - \alpha \lambda_{\max}\{\Lambda\}) \|\dot{\mathbf{e}}\|^2 \\ & -\lambda_{\min}\{\Delta\} \|\alpha \mathbf{e} + \dot{\mathbf{e}}\|_1 \\ & -k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \left[\|\tilde{W}\|_F^2 - \|\tilde{W}\|_F \|W\|_F \right] \\ & + \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \|\boldsymbol{\epsilon}\|_1. \end{aligned} \quad (16)$$

Además, usando el hecho de que

$$\|\tilde{W}\|_F^2 - \|\tilde{W}\|_F \|W\|_F = \left(\|\tilde{W}\|_F - \frac{\|W\|_F}{2} \right)^2 - \frac{\|W\|_F^2}{4},$$

y considerando las siguientes cotas

$$\begin{aligned} \|\boldsymbol{\epsilon}\|_1 &\leq \epsilon_0, \\ \|W\|_F &\leq W_0, \\ \|\mathbf{x}\| &\leq \|\mathbf{x}\|_1, \forall \mathbf{x} \in \mathbb{R}^n, \end{aligned}$$

la ecuación (16) puede reescribirse como

$$\begin{aligned} \dot{L} \leq & -\alpha \lambda_{\min}\{K_p\} \|\mathbf{e}\|^2 + \alpha \lambda_{\max}\{K_d\} \|\mathbf{e}\| \|\dot{\mathbf{e}}\| \\ & -(\lambda_{\min}\{K_d\} - \alpha \lambda_{\max}\{\Lambda\}) \|\dot{\mathbf{e}}\|^2 \\ & -k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \left(\|\tilde{W}\|_F - \frac{W_0}{2} \right)^2 \\ & -\|\alpha \mathbf{e} + \dot{\mathbf{e}}\|_1 \left(\lambda_{\min}\{\Delta\} - \frac{k W_0^2}{4} - \epsilon_0 \right), \end{aligned}$$

que puede expresarse de la siguiente manera

$$\begin{aligned} \dot{L} \leq & -\boldsymbol{\xi}^T Q \boldsymbol{\xi} - k \|\alpha \mathbf{e} + \dot{\mathbf{e}}\| \left(\|\tilde{W}\|_F - \frac{W_0}{2} \right)^2 \\ & -\|\alpha \mathbf{e} + \dot{\mathbf{e}}\|_1 \left(\lambda_{\min}\{\Delta\} - \frac{k W_0^2}{4} - \epsilon_0 \right), \end{aligned} \quad (17)$$

donde $\boldsymbol{\xi} = [\|\mathbf{e}\| \|\dot{\mathbf{e}}\|]^T \in \mathbb{R}^2$ y

$$Q = \begin{bmatrix} \alpha \lambda_{\min}\{K_p\} & -\frac{\alpha}{2} \lambda_{\max}\{K_d\} \\ -\frac{\alpha}{2} \lambda_{\max}\{K_d\} & \lambda_{\min}\{K_d\} - \alpha \lambda_{\max}\{\Lambda\} \end{bmatrix} \in \mathbb{R}^{2 \times 2}.$$

Para garantizar que \dot{L} sea negativa semidefinida, con base en la cota superior (17) la matriz Q debe ser positiva definida por lo cual las siguientes condiciones deben satisfacerse

$$\begin{aligned} \lambda_{\min}\{\Delta\} &\geq \epsilon_0 + \frac{k W_0^2}{4}, \\ \frac{4 \lambda_{\min}\{K_p\} \lambda_{\min}\{K_d\}}{4 \lambda_{\min}\{K_p\} \lambda_{\max}\{\Lambda\} + \lambda_{\max}\{K_d\}^2} &< \alpha < \frac{\lambda_{\min}\{K_d\}}{\lambda_{\max}\{\Lambda\}}. \end{aligned} \quad (18)$$

Entonces, mientras las condiciones en (13) y (18) se satisfagan, $L > 0$ y $\dot{L} \leq 0$, lo que implica que \mathbf{e} , $\dot{\mathbf{e}}$ y \tilde{W} están acotados. Ahora, integrando ambos miembros de (17) se obtiene la siguiente desigualdad

$$\frac{L(t_0)}{\lambda_{\min}\{Q\}} \geq \int_{t_0}^t \left\| \|\mathbf{e}(t)\| \|\dot{\mathbf{e}}(t)\| \right\|^T dt.$$

Una vez obtenido este resultado, por el lema de Barbalat se concluye que los estados $\mathbf{e}(t)$, $\dot{\mathbf{e}}(t) \rightarrow 0$ cuando $t \rightarrow \infty$.

V. RESULTADOS EN SIMULACIÓN NUMÉRICA

Para evaluar la funcionalidad del esquema de control propuesto y sus prestaciones, se ha realizado un análisis comparativo mediante simulaciones numéricas en MATLAB-Simulink utilizando los parámetros correspondientes al quadrotor AR.Drone reportados en [29]

$$\begin{aligned} K_u &= \text{diag} \{4.72 \ 6.23 \ 2.65 \ 2.38\}, \\ K_v &= \text{diag} \{0.28 \ 0.53 \ 2.58 \ 1.52\}, \end{aligned}$$

que fueron empleados para simular el sistema (2).

El controlador tipo PID utilizado para la comparación está dado como

$$\mathbf{u} = [T(\psi) K_u]^{-1} \left[K_1 \mathbf{e} + K_2 \int_0^t \mathbf{e} dt + K_3 \dot{\mathbf{e}} + \ddot{\mathbf{x}}_d^w + K_v \dot{\mathbf{x}}^w \right], \quad (19)$$

donde K_1 , K_2 y $K_3 \in \mathbb{R}^{4 \times 4}$ son matrices diagonales y positivas definidas que contienen las ganancias de las acciones proporcional, integral y derivativa, respectivamente.

El controlador (7) será denotado como CRNA, que es el acrónimo de “controlador basado en redes neuronales adaptables”, y el controlador (19) será denotado como PID. El controlador PID fue diseñado con base en la dinámica del error en (4) y se agregaron elementos para compensar la dinámica del modelo del quadrotor. Cabe resaltar que para que la compensación del modelo sea efectiva, los parámetros del quadrotor y del controlador interno deben ser conocidos.

La tarea asignada es el seguimiento de trayectorias que trazan una lemniscata con una altura variable descrita por las siguiente expresiones

$$\begin{aligned} x_d(t) &= 0.5 \text{sen} \left(\frac{2\pi}{4} t \right) \text{ [m]}, \\ y_d(t) &= \cos \left(\frac{2\pi}{8} t \right) \text{ [m]}, \\ z_d(t) &= 1.0 + 0.4 \text{sen} \left(\frac{2\pi}{4} t \right) \text{ [m]}, \\ \psi_d(t) &= 0.0 \text{ [}^\circ\text{]}. \end{aligned}$$

Además, se agrega una perturbación $\delta(t) = [\delta_x(t) \ \delta_y(t) \ \delta_z(t) \ \delta_\psi(t)]^T$ la cual es nula en intervalo $0 [s] \leq t < 60 [s]$ y

$$\delta(t) = \begin{bmatrix} \delta_x(t) \\ \delta_y(t) \\ \delta_z(t) \\ \delta_\psi(t) \end{bmatrix} = \begin{bmatrix} \text{sen} \left(\frac{2\pi}{4} t \right) \\ \text{cos} \left(\frac{2\pi}{8} t \right) \\ \text{sen} \left(\frac{2\pi}{4} t \right) \\ \text{sen} \left(\frac{2\pi}{4} t \right) \end{bmatrix}, \text{ para todo } t \geq 60 [s].$$

Las unidades de $\delta_x(t)$, $\delta_y(t)$ y $\delta_z(t)$ son $[m/s^2]$ y de $\delta_\psi(t)$ son $[^\circ/s^2]$. Dicha perturbación es incluida con el propósito de mostrar las ventajas del controlador propuesto.

Las ganancias de los controladores CRNA y PID fueron obtenidas mediante prueba y error, resultando en los valores presentados a continuación.

Para el controlador CRNA descrito en (7):

$$\begin{aligned} L_N &= 10, \quad \alpha = 0.725, \quad k = 0.1, \\ K_p &= \text{diag} \{0.1589 \ 0.1204 \ 0.2830 \ 0.3151\}, \\ K_d &= \text{diag} \{0.1589 \ 0.1204 \ 0.2830 \ 0.3151\}, \\ \Delta &= \text{diag} \{0.001 \ 0.001 \ 0.001 \ 0.001\}, \\ N &= \text{diag} \{26 \ 26 \ 26 \ 26\}. \end{aligned}$$

Los elementos de la matriz de pesos de entrada V son valores aleatorios entre 0 y 1.

Para el controlador PID descrito en (19):

$$\begin{aligned} K_1 &= \text{diag} \{2.15 \ 2.15 \ 2.25 \ 2.65\}, \\ K_2 &= \text{diag} \{0.03 \ 0.03 \ 0.04 \ 0.01\}, \\ K_3 &= \text{diag} \{1.05 \ 1.05 \ 1.25 \ 0.95\}. \end{aligned}$$

Cabe mencionar que el controlador PID fue sintonizado buscando obtener los valores mínimos de error durante la tarea de seguimiento de trayectorias. Sin embargo, al ajustar las ganancias para obtener un desempeño competitivo en el intervalo con perturbación, el desempeño en el intervalo sin perturbación se veía comprometido y viceversa. Por tal motivo la sintonización de ganancias utilizada es aquella que otorga el mejor desempeño y garantiza la estabilidad del quadrotor durante la tarea de seguimiento de trayectorias.

La ruta descrita por el quadrotor al realizar la tarea de seguimiento de trayectoria se presenta en la figura 1 donde se observa en color rojo la correspondiente al controlador CRNA y en color azul la que representa al controlador PID. Puede apreciarse que ambos esquemas de control logran realizar el seguimiento de la ruta en el espacio tridimensional.

El seguimiento de las trayectorias $x(t)$, $y(t)$, y $z(t)$ así como la orientación $\psi(t)$ se presentan en la figura 2. Puede observarse que el controlador CRNA mantiene la posición y orientación cercana a la referencia, incluso después de los 60 segundos que es cuando se presentan las perturbaciones.

Los errores de seguimiento de posición y orientación se muestran en la figura 3, donde es posible observar que los errores obtenidos con el controlador CRNA permanecen cercanos a cero una vez que el transitorio ha pasado incluso cuando las perturbaciones se hacen presentes.

Las acciones de control generadas por el controlador CRNA y el controlador PID se muestran en la figura 4. Note que las señales de control son adimensionales.

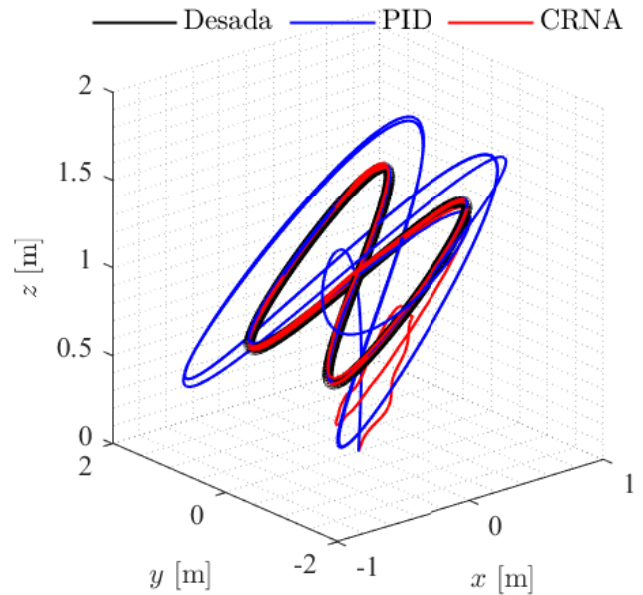


Figura 1. Rutas descritas por el quadrotor en simulación usando los controladores CRNA y PID.

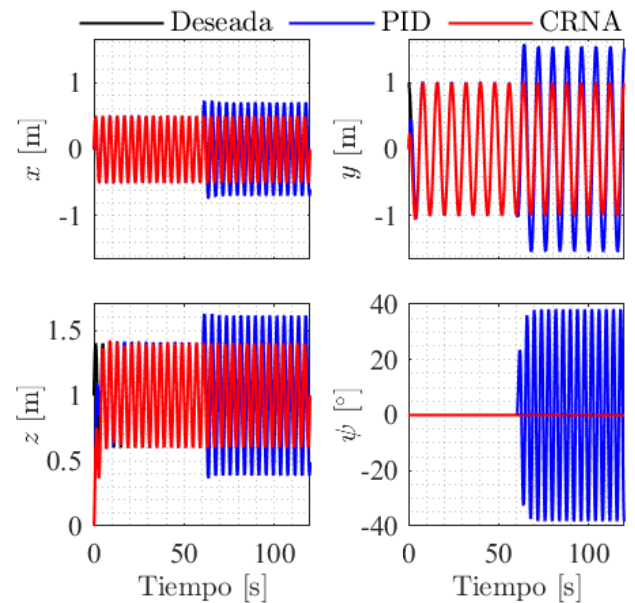


Figura 2. Evolución temporal de la pose del quadrotor $x^w(t)$ en la simulación de la tarea de seguimiento.

Tabla I

 VALOR RMS DEL ERROR $e(t)$ DURANTE LA TAREA DE SEGUIMIENTO DE TRAYECTORIAS EN SIMULACIÓN.

Señal	Sin perturbación			Con perturbación		
	PID	CRNA	(*%)	PID	CRNA	(*%)
\tilde{x} [m]	0.0032	0.0012	63.6	0.4187	0.0010	99.8
\tilde{y} [m]	0.0042	0.0009	79.7	0.4052	0.0022	99.4
\tilde{z} [m]	0.0101	0.0054	46.2	0.3575	0.0025	99.3
$\tilde{\psi}$ [°]	0.0000	2.31×10^{-6}	inf	26.4	0.0660	99.8

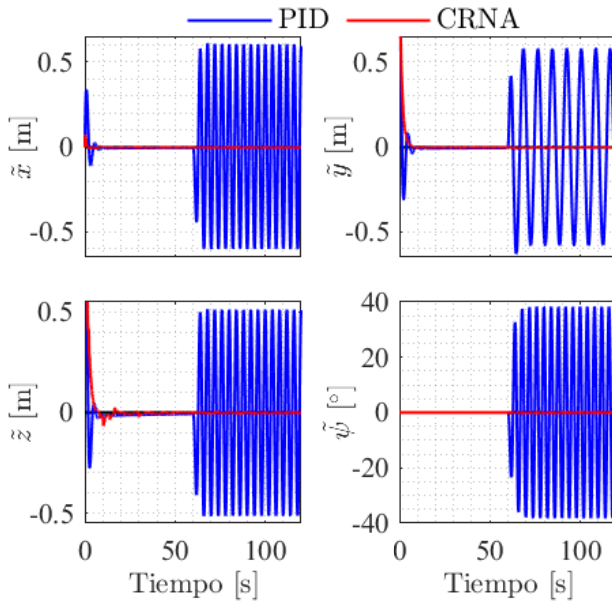


Figura 3. Evolución temporal de los errores de seguimiento en simulación utilizando el controlador CRNA y el controlador PID.

El índice de desempeño seleccionado para la comparación de los esquemas CRNA y PID es el valor cuadrático medio (RMS, por sus siglas en inglés) del error de seguimiento $e(t)$. El valor RMS de los errores de seguimiento se calculó en dos segmentos de tiempo. El primer segmento es $15 [s] \leq t < 60 [s]$ que corresponde al periodo en que el transitorio se ha desvanecido y no se presentan perturbaciones, el segundo segmento es $60 [s] \leq t \leq 120 [s]$ que es donde se presentan las perturbaciones. Además, se calculó el porcentaje de mejora obtenido con el controlador CRNA con respecto del controlador PID. Dichos resultados se presentan en la tabla I. Como puede apreciarse, con el controlador CRNA se obtienen los valores más bajos de error, y los porcentajes de mejora en los intervalos con y sin perturbaciones son satisfactorios.

VI. CONCLUSIONES

En este documento, se presentó un controlador de seguimiento de trayectorias basado en redes neuronales adaptables. La propuesta se desarrolló para un quadrotor con un controlador interno. Se presentó el análisis de convergencia del error con lo que se garantiza la funcionalidad del sistema en lazo cerrado y el acotamiento de los pesos de salida de la red neuronal. Se incluyó un análisis comparativo en simulación utilizando como punto de comparación un controlador PID. Los resultados obtenidos en simulación mostraron que el esquema de control propuesto proveyó de un mejor desempeño en comparación con el controlador PID durante el seguimiento de trayectorias y fue robusto a perturbaciones. Además, representa una opción competente y funcional ya que tiene una estructura simple y no requiere del conocimiento de los parámetros del quadrotor.

REFERENCIAS

- [1] J. del Cerro, C. Cruz Ulloa, A. Barrientos, and J. de León Rivas, "Unmanned aerial vehicles in agriculture: A survey," *Agronomy*, vol. 11, no. 2, p. 203, 2021.
- [2] J. J. Roldán-Gómez, E. González-Gironda, and A. Barrientos, "A survey on robotic technologies for forest firefighting: Applying drone swarms to improve firefighters' efficiency and safety," *Applied Sciences*, vol. 11, no. 1, p. 363, 2021.
- [3] M. A. Olivares-Mendez, C. Fu, P. Ludvig, T. F. Bissyandé, S. Kannan, M. Zurad, A. Annaiyan, H. Voos, and P. Campoy, "Towards an autonomous vision-based unmanned aerial system against wildlife poachers," *Sensors*, vol. 15, no. 12, pp. 31362–31391, 2015.
- [4] S. Lee, D. Har, and D. Kum, "Drone-assisted disaster management: Finding victims via infrared camera and lidar sensor fusion," in *Proc. 3rd Asia-Pacific World Congress on Computer Science and Engineering*, Fiji, Melanesia, December 5–6 2016, pp. 84–89.

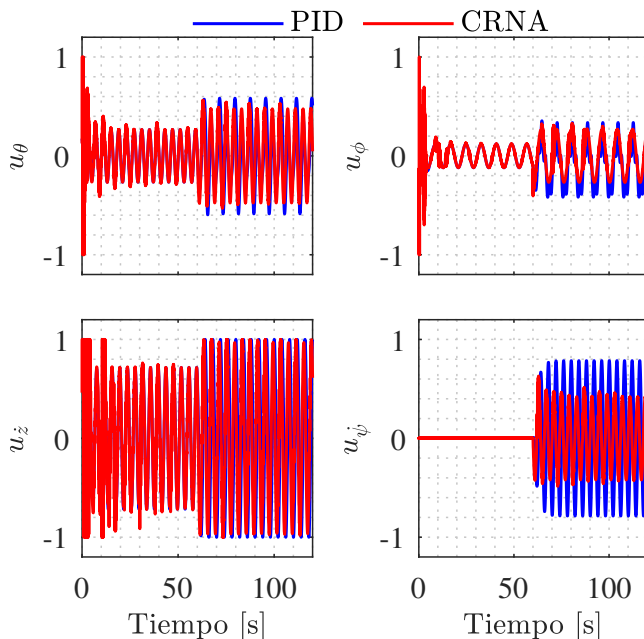


Figura 4. Acciones de control provistas por los controladores en simulación.

- [5] W. Dong, G.-Y. Gu, X. Zhu, and H. Ding, "Development of a quadrotor test bed—modelling, parameter identification, controller design and trajectory generation," *International Journal of Advanced Robotic Systems*, vol. 12, no. 2, p. 7, 2015.
- [6] Z. Zhang and M. Cong, "Controlling quadrotors based on linear quadratic regulator," *Applied Science and Technology*, vol. 5, pp. 38–42, 2011.
- [7] T. Kang, K. J. Yoon, T.-H. Ha, and G. Lee, "H-infinity control system design for a quad-rotor," *Journal of Institute of Control, Robotics and Systems*, vol. 21, no. 1, pp. 14–20, 2015.
- [8] U. Tilki and A. C. Erüst, "Second order sliding mode controller design with optimum characteristic equation for quadrotors," *Mugla Journal of Science and Technology*, vol. 6, no. 2, pp. 150–157, 2020.
- [9] L. Martins, C. Cardeira, and P. Oliveira, "Feedback linearization with zero dynamics stabilization for quadrotor control," *Journal of Intelligent & Robotic Systems*, vol. 101, no. 1, pp. 1–17, 2021.
- [10] A. T. Azar, F. E. Serrano, A. Koubaa, and N. A. Kamal, "Backstepping H-infinity control of unmanned aerial vehicles with time varying disturbances," in *Proc. Int. Conf. of Smart Systems and Emerging Technologies*, Riyadh, Saudi Arabia, Nov. 3–5 2020, pp. 243–248.
- [11] P. Roque, E. Bin, P. Dos Santos Miraldo, and D. V. Dimarogonas, "Fast model predictive image-based visual servoing for quadrotors," in *Proc. Int. Conf. on Intelligent Robots and Systems*, Las Vegas (NV), USA, Oct 24–30 2020.
- [12] R. Pérez-Alcocer and J. Moreno-Valenzuela, "Adaptive control for quadrotor trajectory tracking with accurate parametrization," *IEEE Access*, vol. 7, pp. 53 236–53 247, 2019.
- [13] L. G. Carrillo, A. Dzul, and R. Lozano, "Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3159–3170, 2012.
- [14] A. Al-Mahturi, F. Santoso, M. A. Garratt, and S. G. Anavatti, "Modeling and control of a quadrotor unmanned aerial vehicle using type-2 fuzzy systems," in *Unmanned Aerial Systems*. London, United Kingdom: Academic Press, 2021, pp. 25–46.
- [15] A. R. Dooraki and D.-J. Lee, "An innovative bio-inspired flight controller for quad-rotor drones: Quad-rotor drone learning to fly using reinforcement learning," *Robotics and Autonomous Systems*, vol. 135, p. 103671, 2021.
- [16] R. Adlakha and M. Zheng, "Two-step optimization-based iterative learning control for quadrotor UAVs," *Journal of Dynamic Systems, Measurement, and Control*, 2021.
- [17] D. Zheng and P. Tsiotras, "Near-optimal finite-time feedback controller synthesis using supervised and unsupervised learning," in *American Institute of Aeronautics and Astronautics Scitech Forum*, 2021, p. 1950.
- [18] X. Lin, Y. Wang, and Y. Liu, "Neural-network-based robust terminal sliding-mode control of quadrotor," *Asian Journal of Control*, 2020.
- [19] C. Rosales, F. Rossomando, C. Soria, and R. Carelli, "Neural control of a quadrotor: A state-observer based approach," in *Proc. Int. Conf. on Unmanned Aircraft Systems*, Dallas, TX, USA, 2018, pp. 647–653.
- [20] O. Doukhi and D. J. Lee, "Neural network-based robust adaptive certainty equivalent controller for quadrotor UAV with unknown disturbances," *International Journal of Control, Automation and Systems*, vol. 17, no. 9, pp. 2365–2374, 2019.
- [21] S.-E.-I. Hasseni and L. Abdou, "Adaptive nonlinear robust control of an underactuated micro UAV," *International Journal of Dynamics and Control*, pp. 1–23, 2021.
- [22] V. P. Tran, F. Santoso, M. Garratt, and S. Anavatti, "Neural network-based self-learning of an adaptive strictly negative imaginary tracking controller for a quadrotor transporting a cable-suspended payload with minimum swing," *IEEE Transactions on Industrial Electronics*, 2020.
- [23] Ö. Bingöl and H. Güzey, "Neuro sliding mode control of quadrotor UAVs carrying suspended payload," *Advanced Robotics*, pp. 1–12, 2021.
- [24] L. V. Santana, A. S. Brandao, M. Sarcinelli-Filho, and R. Carelli, "A trajectory tracking and 3D positioning controller for the AR.Drone quadrotor," in *Proc. International Conference on Unmanned Aircraft Systems*, Orlando, FL, USA, May 27–30 2014, pp. 756–767.
- [25] M. C. Santos, L. V. Santana, M. M. Martins, A. S. Brandão, and M. Sarcinelli-Filho, "Estimating and controlling UAV position using RGB-D/IMU data fusion with decentralized information/Kalman filter," in *Proc. IEEE International Conference on Industrial Technology*, Seville, Spain, March 17–19 2015, pp. 232–239.
- [26] M. C. Santos, M. Sarcinelli-Filho, and R. Carelli, "Trajectory tracking for UAV with saturation of velocities," in *Proc. International Conference on Unmanned Aircraft Systems*, Arlington, VA, USA, June 7–10 2016, pp. 643–648.
- [27] C. Rosales, C. M. Soria, and F. G. Rossomando, "Identification and adaptive PID control of a hexacopter UAV based on neural networks," *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 1, pp. 74–91, 2019.
- [28] F. L. Lewis, S. Jagannathan, and A. Yesildirak, *Neural network control of robot manipulators and non-linear systems*. Philadelphia, PA, USA: CRC press, 1999.
- [29] J. A. Sarapura, F. Roberti, J. M. Toibero, J. M. Sebastián, and R. Carelli, "Visual servo controllers for an UAV tracking vegetal paths," in *Machine Vision and Navigation*. Cham, Switzerland: Springer, 2020, pp. 597–625.

Experimentación y simulación de un sistema accesible para implementar algoritmos de control clásicos y discontinuos en motores DC

M. I. Ruiz Gasca Marco Antonio
 Departamento de eléctrica electrónica
 TECNM campus Tláhuac
 Ciudad de México, Alcaldía Tláhuac
 marco.ruiz@tlahuac.tecnm.mx

Resumen—Dadas las limitaciones para acceder a laboratorios de manera presencial, originadas por las restricciones de la emergencia sanitaria en el mundo y específicamente a inicios del año 2020 en México, se propone este trabajo enfocado a la implementación de algoritmos de control en un motor utilizando componentes accesibles. El tipo de dispositivos necesarios o equivalentes están al alcance de muchos estudiantes y académicos. En este documento se presentan los resultados experimentales con sustento teórico para este esquema de implementación, útil en diversos tipos de motores DC para control de velocidad, los algoritmos aplicados son el clásico PID continuo y un Súper-Twisting de la familia de modos deslizantes (algoritmos discontinuos). Los resultados, procedimiento y esquemáticos se detallan en el presente trabajo.

Palabras clave: Control, DC motor, control clasico, Arduino, PID, Sliding Modes, Modos Deslizantes

I. INTRODUCCIÓN

I-A. Motivación y objetivo

Aunque el control de motores de DC es un tema muy conocido y explorado, existe una motivación ligada al actual confinamiento, la contingencia originada por la pandemia del SARS-CoV-2 nos ha obligado a las instituciones educativas a mantenernos alejados de los laboratorios en donde generalmente se encuentra equipo necesario para las prácticas y experimentos. En experimentación requerimos fuentes, procesadores, controladores, tarjetas de desarrollo, etc. además normalmente los dispositivos (DSP, osciloscopios, FPGA, servodrivers) tienen características sofisticadas y de alto costo para el gasto de una persona promedio. Si la calidad de los componentes es menor entonces se acentúa la brecha entre la teoría a la práctica, derivado de tiempos de muestreo altos, baja resolución de sensores, poca potencia disponible (saturación) y ruido de las señales, hacen el trabajo más difícil. Aún con todas estas limitaciones se propone documentar los elementos necesarios para posibles prácticas y en un futuro incorporar el esquema a más proyectos. Explotaremos al máximo estos componentes fáciles de encontrar y asequibles.

El objetivo de control es estabilizar la velocidad de un motor DC, primero con un PID clásico como punto de partida, siendo además la técnica más utilizada en la práctica. También

usaremos un algoritmo por modos deslizantes continuos Súper-Twisting cuyas propiedades de robustez son atractivas.

I-B. Control de motores DC

Dentro del contexto de control de motores de DC existen muchos trabajos donde se abordan diversos problemas: identificación, uso mínimo de sensores, etc. Como se mencionó en la motivación, algunos de estos trabajos están basados en la utilización de equipo que no está al alcance fácilmente, en [9] por ejemplo, se consiguen resultados excelentes pero utilizan equipo sofisticado tanto para adquisición como para control. También existen trabajos donde usan tarjetas de desarrollo similares a la de este trabajo, tal es el caso de [1] entre otros, pero no plantean el uso de controladores por modos deslizantes como en este trabajo. Existen publicaciones que estudian el mismo tema, pero muchas veces solo consideran simulaciones sin enfrentarse a todas las complicaciones que implica el uso experimental, por ejemplo en [2]. En resumen nuestro aporte radica en la propuesta de empezar a utilizar controladores modernos y más sofisticados con dispositivos que estén al alcance de más personas, esto podría resultar en una infinidad de nuevas posibilidades tanto para educación como en aplicaciones de proyectos reales.

I-C. Control discontinuo por modos deslizantes

El control de sistemas inciertos hoy en día es una necesidad práctica. Para resolver el problema, una de las técnicas más prometedoras es el control por modos deslizantes, reportado por el profesor V. Utkin y presente en diversidad de fuentes bibliográficas [10].

Por un lado tenemos los algoritmos de orden 1, sin embargo el problema en implementación sobre actuadores, es el fenómeno de *chattering* [10]. Para dicho problema se puede utilizar el algoritmo Super-Twisting [6], el cuál nos da una señal de control continua y ha sido muy estudiado desde varios enfoques como estabilidad en sentido de Lyapunov [7] además de muchos otros trabajos en años recientes y gracias a ello se conoce mucho acerca de este algoritmo, sin embargo las propiedades de convergencia en tiempo finito se pierden para un sistema de segundo orden. Existen algoritmos de orden 2

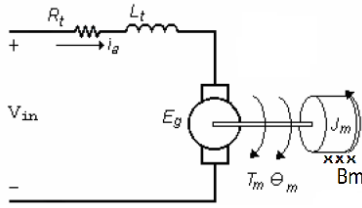


Figura 1. Esquema electromecánico del motor DC para el modelo.

como el Twisting, el cual da convergencia **teórica** en tiempo finito a los estados pero el problema de *chattering* regresa.

Para un sistema mecánico que es de orden dos, se ha propuesto utilizar algoritmos por modos deslizantes de alto orden añadiendo un integrador para suavizar la señal (en el caso de un sistema mecánico lo más adecuado sería utilizar uno de orden 3), pero es necesario el conocimiento de al menos una segunda derivada para implementarlo, en este caso la motivación de utilizar los modos deslizantes pierde sentido, ya que con esa información el problema puede ser resuelto sin la necesidad de modos deslizantes.

Aunque actualmente existen avances muy importantes en el área de modos deslizantes [11], no hay publicaciones reportando como se comportan con estos dispositivos que usaremos en el presente trabajo.

I-D. Abreviaciones y acrónimos

Definiremos algunas abreviaturas usadas constantemente en este trabajo: RPM revoluciones por minuto, DC direct current (corriente continua), PID proporcional integral derivativo, SMC sliding mode control, SISO single input single output, MD modos deslizantes, ST Súper-Twisting, DAC digital analog converter (conversor analógico digital), PPR pulsos por revolución.

I-E. Organización del documento

El documento se organiza en las siguientes secciones: en el modelo matemático fundamentamos el tipo de sistema con el que estaremos trabajando y es la base para las simulaciones, en la sección de algoritmos de control detallamos analíticamente las técnicas utilizadas para controlar el sistema, en la sección de experimentos podemos apreciar las gráficas resultado de la implementación física con detalles de las condiciones de implementación.

II. MODELO MATEMÁTICO DEL MOTOR DE DC

El modelo dinámico del motor de DC con el que trabajaremos está compuesto por una dinámica electromecánica, las variables consideradas son: i_a en amperios [A] para la corriente del circuito, $V_{in}(t)$ es el voltaje de entrada al motor en Volts [V] y $\theta(t)$ es la posición angular del eje del motor, la velocidad angular es su derivada representada por $\omega(t) = \frac{d\theta(t)}{dt}$. Al desarrollar las leyes de cada elemento y en conjunto, llegaremos a las ecuaciones 1 y 2, obtenidas a partir de considerar el diagrama de la Figura 1.

$$V_{in} = R_t i_a + L_t \frac{di_a}{dt} + E_g \quad (1)$$

$$T_m = J_m \frac{d^2\theta}{dt^2} + T_L + B_m \frac{d\theta}{dt} \quad (2)$$

De las ecuaciones 1 y 2 tenemos los siguientes parámetros:

- R_t resistencia total del circuito.
- L_t inductancia total o equivalente del circuito.
- J_m inercia rotacional del motor.
- B_m constante de fricción viscosa.

$$G(s) = \frac{k}{J_m L s^2 + (J_m R_t + L_t B_m) s + B_m R_t + k^2} \quad (3)$$

III. ALGORITMOS DE CONTROL

Se proponen para este trabajo dos esquemas de control, un esquema clásico con resultados bien conocidos ampliamente en la literatura y un esquema discontinuo que bajo estas limitaciones, no tanto. La propuesta también implica la intención de saber experimentalmente si nuestro motor de pruebas será capaz de responder satisfactoriamente con algoritmos más sofisticados y el sistema de prueba es apto para futuros proyectos, en los cuales se vean involucrados mayores retos o esquemas más elaborados.

III-A. Controladores clásicos PID

Los controladores clásicos a usar tendrán la ley de control dada por la expresión 4, de la cual $e(t) = r(t) - \omega(t)$ está definida como el error de velocidad, las constantes k_p , k_i y k_d son las ganancias del controlador que serán sintonizadas.

$$u(t) = k_p e(t) + k_i \int_0^\infty e(t) + k_d \frac{de(t)}{dt} \quad (4)$$

III-B. Controladores discontinuos por modos deslizantes

Para ajustar las ganancias de acuerdo a la metodología del control por modos deslizantes, se considera un doble integrador, con las perturbaciones acopladas a la señal de control. Renombrando la la velocidad x_1 y x_2 como su derivada respectivamente. Consideraremos un enfoque SISO

$$\dot{x}_1 = x_2 \quad (5)$$

$$\dot{x}_2 = u_1 + \delta(\dot{x}, t) \quad (6)$$

Se propone implementar el algoritmo Súper-Twisting 7 el cual es robusto ante perturbaciones Lipschitz, c y b son constantes a sintonizar, $\sigma = x_1 + x_2$ es la superficie deslizante.

$$\begin{aligned} u_1 &= -c|\sigma|^{\frac{1}{2}} \text{sign}(\sigma) + \dot{\phi} \\ \dot{\phi} &= b \text{sign}(\sigma) \end{aligned} \quad (7)$$

El control Super-Twisting provee señal continua, utilizando la superficie de deslizamiento. El algoritmo provee de convergencia en tiempo finito a las superficies y convergencia exponencial a los estados,

IV. SIMULACIONES

Las simulaciones son una parte importante en este trabajo, una simulación correctamente desarrollada tendrá validez en la realidad, así que en primera instancia se propondrá un esquema general simplificado, a partir del cual en adelante consideraremos ajustes necesarios en ruido, muestreo y tipo de controlador.

Los parámetros en la simulación son iguales a las de los experimentos:

- Paso fijo de $20[m.s]$.
- Método de solución: Euler 1.
- Ruido de sensor como señal aleatoria limitada en $[-10,+10]$ con muestro de $20[m.s]$.

IV-A. Validación del modelo

Los parámetros utilizados para reproducir y validar el modelo se presentan en la Tabla I.

Tabla I
PARÁMETROS DEL MODELO DINÁMICO DEL MOTOR

Parámetro	Valor	Unidades
R_t	0.54	$[\Omega]$
L_t	0.023	$[H]$
k	0.0951	$[V/rad/s]$
B_m	0.011125	$[Nm/rad/s]$
J_m	0.003074	$[kg \cdot m^2]$

Aplicando los parámetros a la función de transferencia 3, se simuló una prueba a lazo abierto a entrada escalón de magnitud 6 (representando 6 Volts de la fuente) al motor, obteniendo la gráfica de la Figura 2, con esos resultados el modelo matemático es suficientemente fiel al motor experimental.

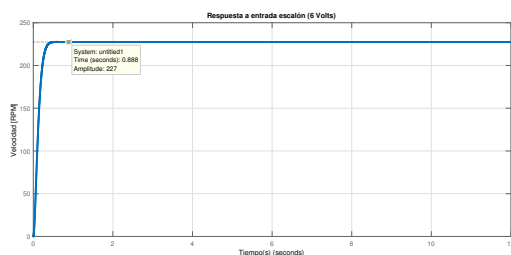


Figura 2. Respuesta al escalón en lazo abierto del modelo planteado.

IV-B. Esquema de control

El primer esquema de control propuesto se puede ver en la Figura 3 donde se muestra la idea general para los experimentos.

IV-C. Simulación PID

Primero se desarrolló la simulación con condiciones ideales, tomando los parámetros del modelo lineal y el diagrama de bloques que se ve en la Figura 4, las gráficas fueron el resultado de implementar el controlador sin ruido, están en la Figura 5.

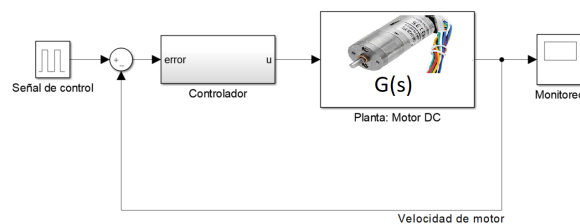


Figura 3. Diagrama de bloques para control de velocidad.

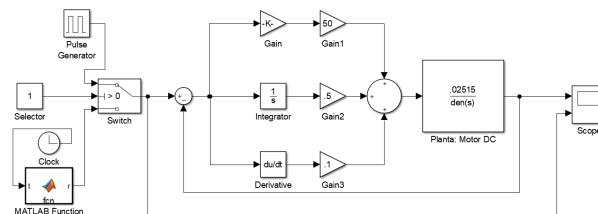


Figura 4. Esquema de simulación para el algoritmo PID.

IV-D. Súper-Twisting

Al implementar el algoritmo Súper-Twisting [6] esperamos anticipar qué sucederá con los resultados en el sistema experimental. El algoritmo Súper-Twisting es un control por modos deslizantes continuo, también tiene propiedades de conmutación a una alta frecuencia, así que sometimos el sistema a dos posibles referencias para verificar, como se muestra en la Figura 7, que el error es desvanecido y la señal de control es continua aunque haciendo uso de un acercamiento (zoom 1 y zoom 2) de la misma Figura 7 es posible verificar que conserva sus propiedades de conmutación, pero cuya amplitud posiblemente no represente un peligro para el sistema físico de experimentación. Una posible elección de las constantes es $\Delta > |\frac{d}{dt}\delta(t)|$ y $k_1 = 1.5\sqrt{\Delta}$, $k_3 = 1.1\Delta$, tales ganancias se pueden verificar en [6].

V. EXPERIMENTOS

Las condiciones físicas de los experimentos implementados son las siguientes:

- Tiempo de muestreo: $20[m.s]$.
- Señal de control: DAC de 12 bits.
- Encoder con resolución de 238 PPR.
- Relación de engranaje 34 : 1.

V-A. Equipo utilizado y diagramas

Los diagramas de conexión para implementar los experimentos se ven en las Figuras 10 y 11, los dispositivos utilizados para este trabajo se enlistan a continuación:

- Motor de DC con encoder 210 RPM, $6[V]$ voltaje nominal, $0.13[A]$ sin carga. Modelo CHIHAI MOTOR 6V 100/210/300RPM Encoder Motor GM25-370 DC Gear Motor.
- Fuente de poder ajustable (también puede ser fija) 0 a 12 Volts.

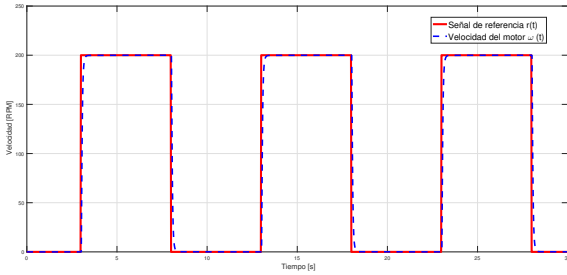


Figura 5. Resultados de simulación control PID sin ruido.

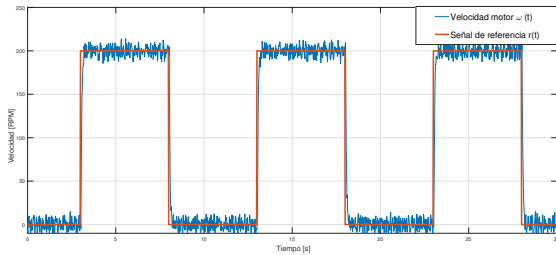


Figura 6. Simulación control PID con ruido.

- Transistor 2N222 y resistor 220 $[\Omega]$.
- Tarjeta de desarrollo Arduino DUE.

V-B. Relación Voltaje-Velocidad

Los primeros experimentos se reportan en la Figura 12, en lazo abierto sin implementar una retroalimentación se identifica la respuesta de velocidad del motor ante un conjunto de voltajes inyectados.

V-C. Controlador clásico PID

En esta subsección se presentan los resultados de implementar 3 referencias constantes al motor con el algoritmo clásico, primero con un controlador proporcional P, después un esquema proporcional integral PI y finalmente el controlador proporcional integral derivativo PID. Las ganancias utilizadas para la implementación de este algoritmo fueron: $k_p = 0.8$, $k_d = 0.3$ y $k_i = 0.3$. En la Figura 13 se pueden apreciar los datos obtenidos al implementar este algoritmo bajo estas ganancias, es posible observar que la señal de control (rojo) es alcanzada por la respuesta del motor (azul) de una forma aceptable aunque con un ligero alejamiento y variaciones en la lectura de velocidad.

V-D. Implementación experimental Súper-Twisting

Para la implementación del controlador Súper-Twisting en el sistema experimental conformado por la tarjeta de desarrollo y el motor, conservamos las mismas condiciones de muestreo, sensor y rutina de control. A excepción claro del algoritmo que estabiliza la velocidad del motor, para el cual satisfactoriamente vimos resultados prometedores, una respuesta suave sin que el ruido sea una limitante que afecte el

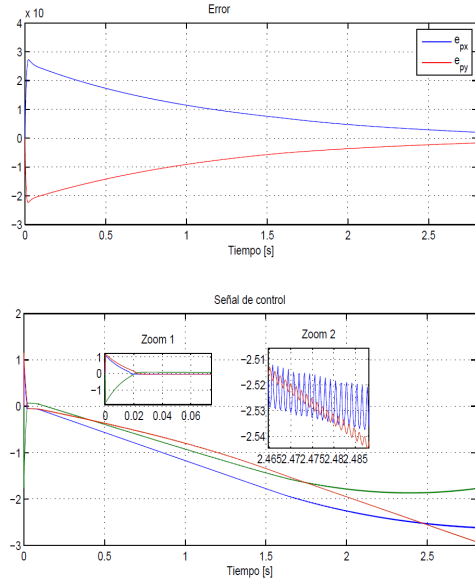


Figura 7. Resultados de simulación algoritmo Súper-Twisting, errores llevados a cero (arriba) y acercamiento para ver la señal discontinua (abajo).

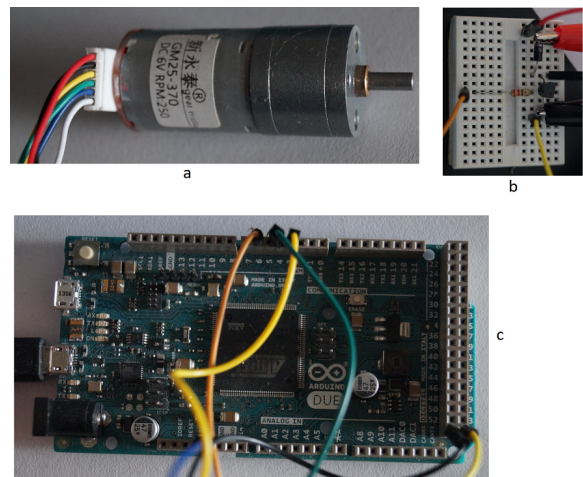


Figura 8. Elementos principales para experimentación física, a: motor, b: etapa de potencia (transistor y resistor), c: tarjeta de desarrollo.

desempeño del controlador. Las gráficas de los datos obtenidos pueden apreciarse en la Figura 15.

El fragmento del código que se desarrolló para implementar este algoritmo se puede observar en la Figura 14, del cual la derivada del error se calculó diferenciando y la integral de ϕ con una suma acumulada en el bucle principal, también para el *sign* parte esencial del controlador discontinuo se utilizó una función dentro del programa principal que con *if* calcula los posibles valores 0, -1 ó 1. La sintonización del algoritmo Súper-Twisting fue más sencilla gracias a la metodología empleada sintonizando sólo el parámetro Δ , las ganancias aplicadas fueron $k_1 = 0.3$ y $k_2 = 0.08$.

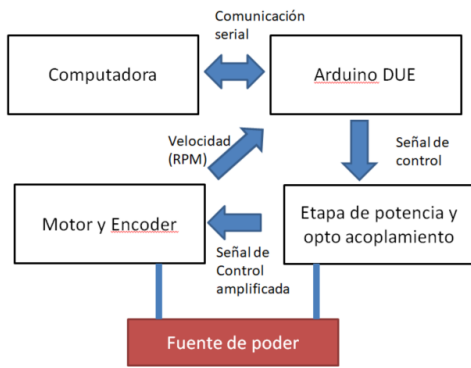


Figura 9. Equipo completo utilizado para el experimento y el flujo de las señales.

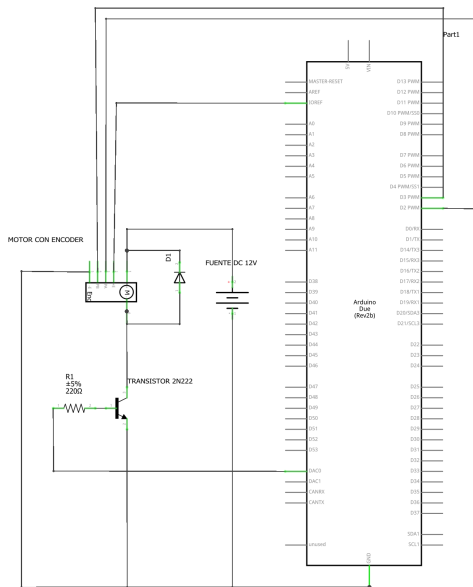


Figura 10. Esquemático de conexiones.

CONCLUSIONES

Probamos satisfactoriamente los algoritmos de control, documentamos con detalle el procedimiento necesario para realizar esta implementación desde la modelación, simulación, propuesta de algoritmos, hasta la experimentación. Con este trabajo ofrecemos una metodología que puedan usar estudiantes y académicos, los experimentos utilizan mucho menos componentes de los usuales para este tipo de trabajos. Planteamos el escenario para implementar técnicas modernas como los Modos Deslizantes, empezando con el Súper-Twisting y planeamos en futuro usar más técnicas de este tipo como diferenciadores o algoritmos de alto orden, ganancias adaptativas, autosintonizables, etc. Cabe destacar que abordamos el problema con un análisis sustentado a partir de estas condiciones: tiempo de muestro, perturbaciones e incertidumbres, resolución baja en actuadores

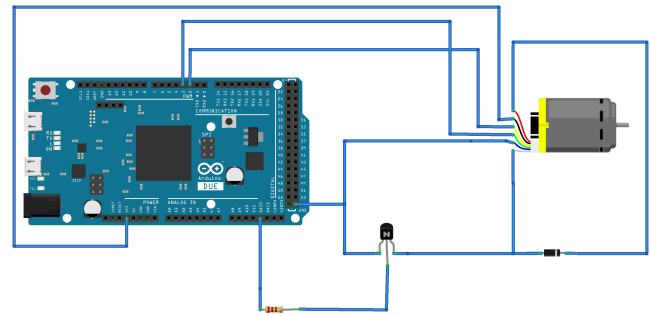


Figura 11. Esquemático de conexiones (vista de componentes).

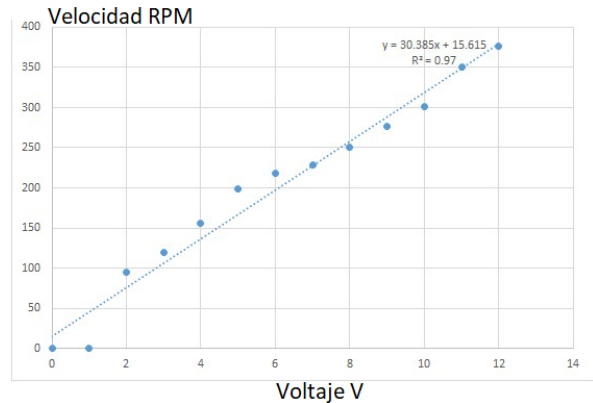


Figura 12. Relación velocidad contra voltaje, identificación de la zona de operación para el motor.

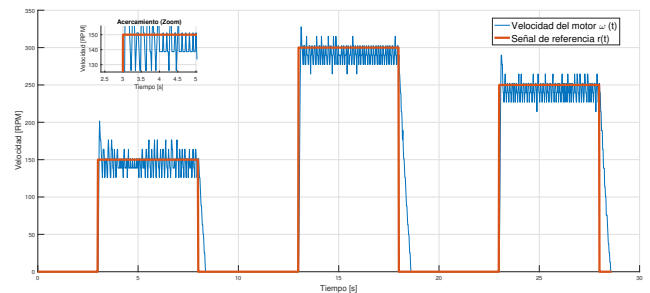


Figura 13. Resultados experimentales de implementar el controlador PID al motor.

```

1 float ST=k1*(pow(abs(sigma),0.5))*sgn(sigma)+phi
2 ;
3 if (ST>4095){ST=4095;}
4 if (ST<0){ST=0;}
5 ST=(int) ST;
   analogWrite (DAC0, ST);

```

Figura 14. Fragmento de código empleado para implementar experimentalmente el Súper-Twisting.

y ruido en sensores.

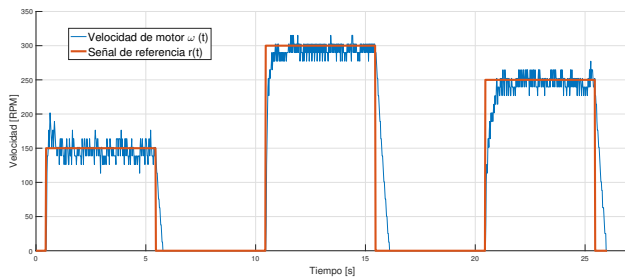


Figura 15. Súper-Twisting, resultados de implementación experimental.

El resultado más prometedor es el obtenido tras implementar físicamente el Súper-Twisting, ahora que contamos con validez experimental de su efectividad en este sistema, nos motiva a trabajar con más técnicas y esquemas de mayor sofisticación.

REFERENCIAS

- [1] Alfian Ma'arif, Phisca Aditya, Iswanto, Ahmad Raditya, Nia Maharani, Aninditya Anggari (2020). "Control of DC Motor Using Proportional Integral Derivative (PID): Arduino Hardware Implementation" 2nd International Conference on Industrial Electrical and Electronics (ICIEE).
- [2] Balamurugan S., Umarani A. (2020). "Study of Discrete PID Controller for DC Motor Speed Control Using MATLAB" 2020 International Conference on Computing and Information Technology, University of Tabuk, Kingdom of Saudi Arabia. Volume: 01, Issue: ICCIT- 1441, Page No.: 358 - 363, 9th & 10th Sep. 2020.
- [3] Chapman, S. J. (2012). "Máquinas eléctricas" Ciudad de México: McGraw-Hill.
- [4] Isidori, A. (1995). "Nonlinear Control Systems". Nueva York: Springer Verlag.
- [5] Khalil H. (2014) "Nonlinear Systems" Nueva York: Pearson Education Limited.
- [6] Levant A. (1993) "Sliding order and sliding accuracy in sliding mode control." International Journal of Control, 58, 1247-1263.
- [7] Moreno J. A., Osorio M. (2008) "A Lyapunov approach to second-order sliding mode controllers and observers." Proceedings of the 47th IEEE Conference on Decision and Control Cancun, México.
- [8] Ogata K., "Ingeniería de control moderna" Prentice Hall 5a ed.
- [9] Ramirez-Betancour R., Valenzuela-Murillo F., Martínez-Solís F., Castañeda-Hernández C., Morfin-Garduño A., Olmos-Lopez J., (2018), "Control de velocidad de un motor de CD basado en mediciones de la corriente de armadura" Ingeniería Investigación y Tecnología volumen XIX número 4 ISSN 2594-0732 FI-UNAM.
- [10] Utkin, V., Guldner, J., y Shi, J. (2009), "Sliding Mode Control in Electro-Mechanical Systems" US: Taylor & Francis.
- [11] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant (2014) "Sliding Mode Control and Observation". New York, USA: Birkhauser, 2014.

Sintonía con PSO de un controlador PID para la regulación de posición de una articulación motriz pendular

David Alonso Pimienta Berrelleza
Tecnológico Nacional de México
 Instituto Tecnológico de La Paz
 La Paz, B.C.S., México
 M19310016@lapaz.tecnm.mx

Jesús Alberto Sandoval Galarza
Tecnológico Nacional de México
 Instituto Tecnológico de La Paz
 La Paz, B.C.S., México
 jesus.sg@lapaz.tecnm.mx

Marco Antonio Castro Liera
Tecnológico Nacional de México
 Instituto Tecnológico de La Paz
 La Paz, B.C.S., México
 marco.cl@lapaz.tecnm.mx

Iliana Castro Liera
Tecnológico Nacional de México
 Instituto Tecnológico de La Paz
 La Paz, B.C.S., México
 iliana.cl@lapaz.tecnm.mx

Resumen — En este trabajo se presenta una aplicación del método de optimización por enjambre de partículas (PSO) para la sintonía de un controlador PID aplicado al control de posición de una articulación motriz pendular acoplada a un servomotor. La novedad es la definición de la aptitud, como una sumatoria de índices de desempeño que incluye la magnitud del voltaje del motor, a diferencia de otros trabajos que no lo incluyen. Una ventaja del esquema propuesto es la adecuada selección de ganancias del controlador que permiten el cumplimiento del objetivo de control de regulación de posición, sin exceder el voltaje nominal del motor. Para ilustrar el desempeño del controlador aplicando el esquema de sintonía, se presentan resultados de simulación y experimentales con un servomotor de la marca Quanser.

Palabras clave — Método PSO, regulación de posición, control PID, par, SRV02 Quanser.

I. INTRODUCCIÓN

Entre los métodos más populares para la sintonía de controladores podemos encontrar los siguientes: Ziegler-Nichols, Cohen-Coon, Lopez et al., Sung et al., Chien-Hrones-Reswick, entre otros [1]. De los métodos anteriores, la selección de las ganancias de cada controlador se basa en la experiencia del usuario. Sin embargo, en algunos casos su aplicación no es práctica, por ejemplo, cuando se desconoce la función de transferencia o cuando no se cuenta con el tiempo suficiente para su aplicación [2]. A pesar de ello, existen alternativas como los métodos heurísticos, los cuales ofrecen soluciones adecuadas, en un tiempo computacional aceptable [3].

Entre las alternativas para la sintonía de controladores mediante métodos no convencionales, donde el problema de sintonía es visto como un problema de optimización que presenta limitaciones, una de estas alternativas es el uso de métodos metaheurísticos como PSO [4].

El PSO es un método diseñado para encontrar mínimos y máximos globales de una función de aptitud que puede tener múltiples índices de desempeño. Este es capaz de optimizar un problema mediante la generación de un enjambre de partículas donde cada partícula representa una solución candidata. Las partículas tienen una posición \vec{k} y una velocidad \vec{v} , con la cual se ubica y se mueve, respectivamente, dentro del espacio de búsqueda [5]. Cada partícula es afectada en su movimiento a partir de la mejor posición encontrada de manera individual como de manera grupal, esto con el fin de lograr converger en las posiciones cercanas a la óptima. Cuando esto es aplicado al PID se puede entender que cada solución candidata contiene el grupo de ganancias para el controlador.

Este artículo propone realizar la sintonía de un controlador PID mediante el uso de PSO con un esquema para la aptitud donde se considera la magnitud de voltaje del motor con el fin de lograr un buen desempeño, que cumpla con el objetivo de control sin sobrepasar el voltaje nominal. Esto permitiría, por un lado, evitar que por una mala selección de las ganancias del controlador se exceda el voltaje nominal, y por otro lado, se cumplieran índices de desempeño (integral del error de posición ponderado en el tiempo, integral del sobre-impulso de posición, integral del sobre-impulso de voltaje, etc.) dados por el usuario sobre la respuesta del sistema.

El resto del trabajo está organizado como sigue: en la Sección 2 se describe el modelo dinámico de la articulación motriz pendular, en la Sección 3 se describe el método PSO y se establecen los índices de desempeño, en la Sección 4 se muestran los resultados de simulación y experimentales, y finalmente en la Sección 5 se muestran las conclusiones del trabajo.

II. ARTICULACIÓN MOTRIZ PENDULAR

La articulación motriz pendular consta de un servomotor cuyo eje está conectado en uno de sus extremos a una barra de carga, y esta a su vez a un disco de carga, como se muestra en la figura 1, para la cual el significado de las variables y parámetros se muestra en la tabla I. El motor de CD a partir de un voltaje de entrada genera un par τ , el cual está relacionado con la posición angular q .

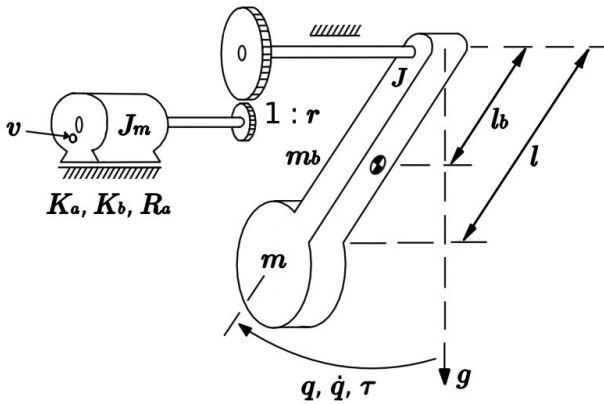


Figura 1. Articulación motriz pendular (motor de CD+péndulo)

 Tabla I
 VARIABLES Y PARÁMETROS DE LA ARTICULACIÓN MOTRIZ PENDULAR

Símbolo	Descripción	Unidades
R_a	Resistencia de armadura	Ω
K_a	Constante motor-par	Nm/A
K_b	Constante de contrarreacción electromotriz	Vs/rad
r	Relación de reducción de engranes (en general $r > 1$)	
v	Voltaje de armadura	V
τ	Par en el eje de la carga	Nm
q	Posición angular del eje de la carga	rad
\dot{q}	Velocidad angular del eje de la carga	rad/s
\ddot{q}	Aceleración angular del eje de la carga	rad/s ²
J_m	Inercia del rotor	kg m ²
J	Inercia del brazo	kg m ²
m_b	masa del brazo	kg
m	masa de la carga	kg
l_b	distancia desde el eje de giro al centro de masa del brazo (sin carga)	m
l	distancia desde el eje de giro a la carga m	m
g	aceleración de la gravedad	m/s ²

II-A. Modelo dinámico

Como se busca regular la posición angular del eje de carga de la articulación motriz pendular, se seleccionó el error de posición angular \tilde{q} como la variable de realimentación para el controlador del sistema en el lazo cerrado, donde el error de posición angular \tilde{q} y su derivada temporal $\dot{\tilde{q}}$ están dadas por las siguientes dos definiciones [6]:

$$\tilde{q} = q_d - q \quad (1)$$

$$\dot{\tilde{q}} = -\dot{q} \quad (2)$$

Nótese que para el caso de regulación de posición q_d es constante para todo $t \geq 0$, por tanto, $\dot{q}_d = 0$.

A partir de las ecuaciones que representan un modelo dinámico de un robot manipulador de n grados de libertad (g.d.l.), un motor de CD ideal y una ecuación que contemple la relación de engranes y la conexión del péndulo al motor, es posible establecer el modelo dinámico del sistema, donde se representa la relación del voltaje de entrada v con la posición angular obtenida en el eje de carga q , dando como resultado el siguiente par de ecuaciones [7]:

El modelo dinámico completo de la articulación motriz pendular puede ser escrito como:

$$\frac{d}{dt} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ \frac{1}{J_m} \left[\frac{K_a}{rR_a} v - \frac{K_a K_b}{R_a} \dot{q} \right] \end{bmatrix} \quad (3)$$

donde el voltaje v , después de algunas manipulaciones algebraicas, está dado por:

$$v = \frac{rR_a}{K_a} \left[[M^{-1}J_m] [\tau - g(q)] + \frac{K_a K_b}{R_a} \dot{q} \right] \quad (4)$$

donde

$$g(q) = [m_b l_b + ml]g \sin(q), \quad (5)$$

y

$$M = J + ml^2 \quad (6)$$

Nótese que hemos omitido la inductancia de armadura L_a , al igual que la fricción en el soporte del brazo f_L y la fricción del rotor f_m , por tanto: $L_a = 0$, $f_L = 0$ y $f_m = 0$.

II-B. Ley de control y objetivo de control

La ley de control PID (Proporcional, Integral y Derivativo) está representada por la siguiente ecuación:

$$\tau = K_p \tilde{q} + K_v \dot{\tilde{q}} + K_i \xi \quad (7)$$

donde

$$\dot{\xi} = \tilde{q} \quad (8)$$

Para este trabajo se estableció arbitrariamente como referencia q_d las posiciones deseadas de 90° (1.571 [rad]) y 45° (0.785 [rad]). El objetivo de control es seleccionar apropiadamente las ganancias K_p , K_v y K_i en (7) tal que logre llevar la posición angular q a la posición deseada q_d . Formalmente el objetivo de control anterior se expresa de la siguiente manera:

$$\lim_{t \rightarrow \infty} |q(t) - q_d| = 0 \quad (9)$$

II-C. Método de sintonía basado en Kelly y Santibáñez (2003)

Basados en el método de sintonía introducido en [7] los valores de K_p , K_v y K_i del controlador PID (7)-(8) deben satisfacer las siguientes tres desigualdades:

$$\lambda_{Max}\{K_i\} \geq \lambda_{min}\{K_i\} > 0, \quad (10)$$

$$\lambda_{Max}\{K_p\} \geq \lambda_{min}\{K_p\} > k_g, \quad (11)$$

$$\lambda_{Max}\{K_v\} \geq \lambda_{min}\{K_v\} > \frac{[\lambda_{Max}\{K_i\}][\lambda_{Max}^2\{M\}]}{[\lambda_{min}\{K_p\} - k_g][\lambda_{min}\{M\}]} \quad (12)$$

donde k_g se calcula de la siguiente manera:

$$k_g = n \left[\text{Max}_{i,j,q} \left| \frac{\partial g_i(q)}{\partial q_j} \right| \right] \quad (13)$$

Debido a que en este trabajo se trata de un servomecanismo de 1 g.d.l. las matrices K_p , K_v y K_i que representan al controlador contienen un solo elemento cada una, por lo cual las ecuaciones (10), (11), (12) y (13) se pueden reescribir como simples escalares:

$$k_i > 0, \quad (14)$$

$$k_p > k_g, \quad (15)$$

$$k_v > \frac{k_i[M^2]}{[k_p - k_g]M}, \quad (16)$$

$$k_g = [m_b l_b] g. \quad (17)$$

Enseguida se presenta un procedimiento de sintonía en el que las ganancias k_p , k_v y k_i son seleccionadas inicialmente de manera arbitraria, pero cumpliendo con (14), (15) y (16). Posteriormente, mediante prueba y error, se modifican las ganancias con el fin de reducir el tiempo de establecimiento y el sobre-impulso. Usando este procedimiento heurístico se obtuvieron los resultados mostrados en la tabla II.

Tabla II
PROPUESTA DE SINTONÍA BASADO EN KELLY Y SANTIBÁÑEZ (2003)

Ganancias del controlador PID (7) para cada posición deseada		
q_d	90°	45°
k_p	0.75	1
k_v	0.09	0.1
k_i	0.21	0.69

III. SINTONÍA DEL CONTROLADOR MEDIANTE PSO

La figura 2 muestra el diagrama de flujo del PSO. Se utilizaron los siguientes valores para los parámetros del PSO: los coeficientes de confianza $c_1 = c_2 = 1.62$ y el peso inercial $w = 0.8$. Su selección se hizo de acuerdo a los resultados satisfactorios mostrados en otro trabajo [8], [9]. La velocidad con la que inicializan las partículas es igual a cero, y se utilizó un tamaño de enjambre de 20 partículas. Las propuestas de sintonía se encontraron mediante pruebas donde se utilizó una configuración de 1000 iteraciones.

Los rangos de creación y de exploración de las partículas para cada una de las posiciones deseadas, fueron definidos en base a los resultados obtenidos en la subsección II-B. Para el caso con $q_d = 90^\circ$ (1.571 rad) el rango de exploración fue de 0 a 0.75, mientras que para $q_d = 45^\circ$ (0.785 rad) fue de 0 a 1. El tiempo de muestreo utilizado para los dos casos mencionados es de 10 segundos y el paso de integración es 0.0001 segundos.

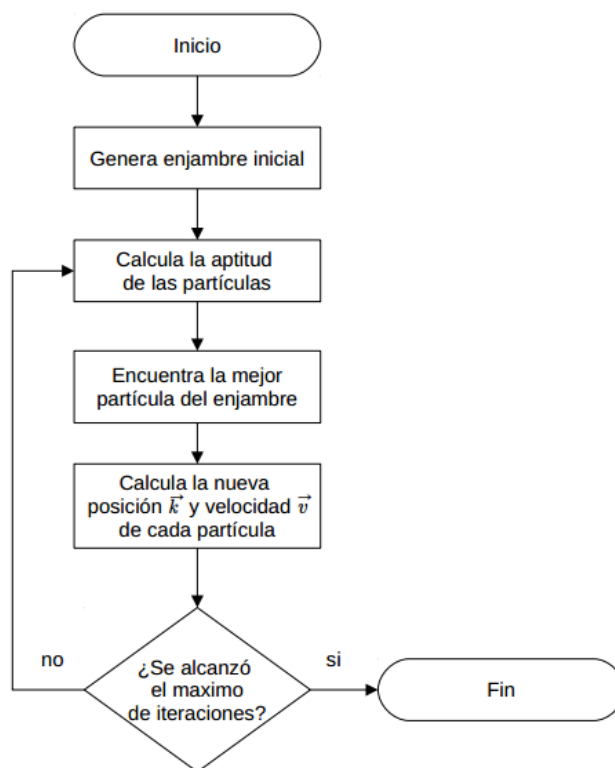


Figura 2. Diagrama de flujo del PSO

III-A. Aptitud e índices de desempeño

Una vez creado el enjambre (recordando que cada partícula representa una solución candidata, que contiene ganancias para el controlador PID), se procede con el cálculo de la aptitud de las partículas. Para ello se establecieron los siguientes índices de desempeño:

Integral del error de posición ponderado en el tiempo, abreviado como *ISTE*: Este error se define como la diferencia

entre la posición angular deseada q_d y la posición angular obtenida q_t en el instante t [6]. Es decir, este índice de desempeño evalúa la posición de la barra de carga q_t otorgando una calificación proporcional a la diferencia entre esta y la posición deseada q_t , tal que entre mayor es la diferencia, mayor será el valor de $ISTE$. Nótese que entre mayor sea la cantidad de tiempo transcurrido t la diferencia entre q_d y q_t tendrá una mayor valoración, misma razón por la que se decidió que al instante t le fuera sumada una unidad, con el fin de que la diferencia de posiciones se valorara desde el primer instante. Este se calcula mediante:

$$ISTE = \int_0^{\infty} (t+1)(q_d - q_t)^2 dt \quad (18)$$

Integral del sobre-impulso de posición IO : Se define como la diferencia entre la posición angular deseada q_d , y la posición angular obtenida q_t en el instante t , cuando la posición angular obtenida q_t es mayor a la posición angular deseada q_d . En otras palabras, este índice de desempeño aumentara de valor cuando la posición angular de la barra de carga sobrepase la posición angular deseada q_d . Este se calcula mediante:

$$IO = \int_0^{\infty} o(q_t) dt \quad (19)$$

siendo $o(q_t)$

$$o(q_t) = \begin{cases} 0, & \text{si } q_d > q_t \\ \alpha_o(q_t - q_d), & \text{en caso contrario} \end{cases} \quad (20)$$

donde α_o es un factor de peso establecido por el usuario de manera arbitraria e indica la severidad con la que se evalúa el índice de desempeño. Para este trabajo $\alpha_o = 100$.

Integral del sobre-impulso de voltaje IV , se calcula mediante (21). Se define como la diferencia entre el voltaje nominal del motor de CD v_n y la magnitud del voltaje calculado antes del limitador de voltaje v_t , cuando la magnitud del voltaje calculado es mayor a la nominal v_n , es decir, este índice de desempeño aumentará de valor cuando el voltaje calculado sobrepase el voltaje con el cual el equipo puede operar de manera segura. Este se calcula mediante:

$$IV = \int_0^{\infty} s(v_t) dt \quad (21)$$

siendo $s(v_t)$

$$s(v_t) = \begin{cases} 0, & \text{si } \alpha_p v_n > v_t \\ \alpha_v(v_t - \alpha_p v_n), & \text{en caso contrario} \end{cases} \quad (22)$$

donde α_v es un factor de peso establecido por el usuario de manera arbitraria e indica la severidad con la que se evalúa el índice de desempeño, y α_p es un factor de tolerancia que multiplica al voltaje nominal v_n e indica la proximidad de la magnitud del voltaje calculado v_t al voltaje nominal v_n , donde un $\alpha_p = 1$ permitiría que v_t pueda igualar el valor de v_n , o donde un $\alpha_p = 0.5$ permitiría que v_t pueda alcanzar hasta el 50 del valor de v_n . Para este trabajo se seleccionó arbitrariamente $\alpha_v = 100$ y $\alpha_p = 0.99$ (99%).

A diferencia de otros trabajos [2], [9], [10], en los que el valor de la aptitud de las soluciones candidatas es dada por un índice de desempeño, u otros [1], [11], [12] donde se suele omitir la magnitud del voltaje y el voltaje nominal dentro del cálculo de la aptitud, en este trabajo, la aptitud calculada usando PSO, es el resultado de sumar los tres índices definidos en (18), (19) y (21), y se calcula mediante (23).

$$A(PID) = ISTE + IO + IV \quad (23)$$

La ventaja de utilizar la integral del sobre-impulso de voltaje IV como un componente del cálculo de la aptitud $A(PID)$ es que en la evaluación de las partículas del PSO se puede penalizar aquellas que en su cálculo de voltaje v_t sobrepasen el voltaje nominal v_n , lo que permite la búsqueda de soluciones con un v_t menor a v_n .

III-B. Propuesta de sintonía PSO

Después de realizar 10 ejecuciones del algoritmo PSO, de las soluciones candidatas obtenidas, se seleccionó aquella que presentó la mejor aptitud para cada uno de los dos casos y estas son tomadas como propuestas de sintonía. Las ganancias y la aptitudes $A(PID)$ de las propuesta de sintonía resultantes de este procedimiento se muestran en la tabla III.

Tabla III
PROPUESTAS DE SINTONÍA MEDIANTE PSO

Ganancias del controlador PID (7) para cada posición deseada		
q_d	90°	45°
k_p	0.459	0.957
k_v	0.057	0.085
k_i	0.296	0.524
$A(PID)$	2539.203	403.616

IV. RESULTADOS

El software utilizado para la simulación y el experimento es MATLAB R2012 en su versión para Windows, con el complemento de Simulink y las librerías de Quanser correspondientes para el SRV02. La configuración utilizada fue la siguiente: El tiempo de simulación utilizado fue de 85 segundos, el método numérico seleccionado fue Runge-Kutta de paso de integración de 0.01 segundos.

Para evaluar el funcionamiento de un controlador PID suelen tomarse en cuenta algunas de las siguientes propiedades: error en estado estacionario, sobre-impulso, tiempo de subida y tiempo de establecimiento [9], [13]. A diferencia de otros trabajos se decidió agregar la magnitud del voltaje (a la salida del controlador) y el par motor como criterios para la evaluación de la respuesta, esto con el fin de obtener más información que permita contrastar el resultado obtenido mediante los métodos de sintonía presentes en este trabajo.

Para las simulaciones y los experimentos, los parámetros del motor de CD de la articulación motriz pendular se extrajeron del manual de usuario del SRV02 de QUANSER [16], con excepción de: $g = 9.81$ [m/s]² y $l_b = 0.045$ [m], este último obtenido simplemente por medición. Con base en el mismo

manual, se agregó un limitador de voltaje, con el objetivo de no sobrepasar el voltaje nominal (6 [V]) a la entrada de la alimentación del motor.

Las tablas y figuras que se muestran en las subsecciones IV-A y IV-B contienen los resultados de cuatro evaluaciones donde se utilizaron las ganancias para el PID mostradas en las tablas II y III. Para facilitar la visualización de los resultados se decidió expresar q_d en grados para ambos casos. Los resultados consideran un "tiempo de establecimiento" en el rango de $q_d \pm 2\%$, y el "Tiempo de subida" desde el 10% hasta 90% tomando como valor final q_d [13]. De acuerdo a las tablas IV-VII, un valor menor significa un mejor resultado en las evaluaciones. Los porcentajes de diferencia mostrados posteriormente a las tablas se han calculado mediante la ecuación (24) donde a y b son los valores numéricos de los criterios de desempeño de cada método de sintonía, tal que a tomaría los correspondientes al método de sintonía basado en Kelly y Santibáñez (2003), mientras que b tomaría los de la sintonía PSO.

$$\frac{(a - b)}{(a + b)/2} \times 100\% \quad (24)$$

Se puede entender que cuando el porcentaje de diferencia es representado con un valor negativo, en ese apartado el método de sintonía basado en Kelly y Santibáñez (2003) obtuvo un mejor resultado en comparación al método de sintonía PSO, en caso contrario, en ese apartado el método de sintonía PSO obtuvo un mejor resultado en comparación al método de sintonía basado en Kelly y Santibáñez (2003). Específicamente en el cálculo de los porcentajes de diferencia de los criterios de desempeño llamados "sobre-impulso (grados)" y "error en estado estacionario (grados)" se optó por sumar el valor de q_d en grados a cada criterio debido a la dificultad que provocaría obtener porcentajes cuando algún valor es igual a 0, p. ej. tomando los valores del sobre-impulso de la tabla IV se obtendría lo siguiente (25):

$$\frac{(0 + 95) - (0.112 + 90)}{[(0 + 95) + (0.112 + 90)]/2} \times 100\% = -0.1 \quad (25)$$

De la ecuación (25) se puede entender que en ese apartado el método de sintonía basado en Kelly y Santibáñez (2003) obtuvo un mejor resultado en comparación al método de sintonía PSO en un 0.1%, tal como se muestra en la figura 5.

IV-A. Simulaciones numéricas

A continuación se muestran los resultados de las simulaciones de regulación de posición de la articulación motriz pendular cuando $q_d = 90^\circ$, donde las figuras 3 y 4 representan las respuestas del sistema, en la primera se observa el registro completo de los 85 segundos de muestreo de q y en la segunda se muestra un acercamiento a los primeros 2 segundos del muestreo de q . La tabla IV y la figura 5 representan las evaluaciones de la respuesta del sistema y la comparativa de las respuestas entre los dos métodos de sintonía respectivamente. De igual manera para el caso donde $q_d = 45^\circ$, las figuras 6 y 7 representan las respuestas del sistema, mientras la tabla V

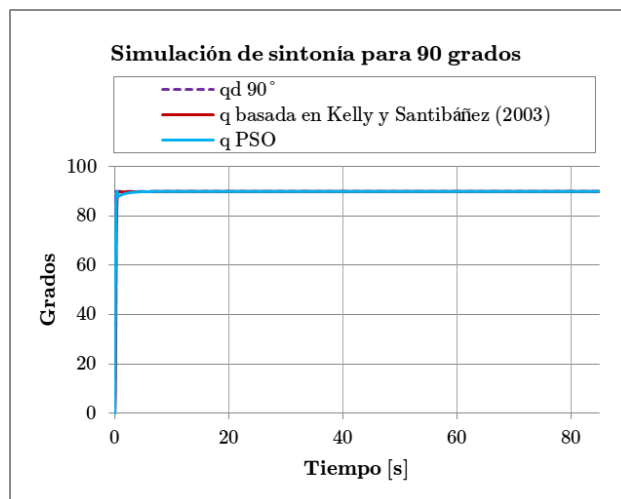


Figura 3. Simulación de sintonía para 90° (85 segundos)

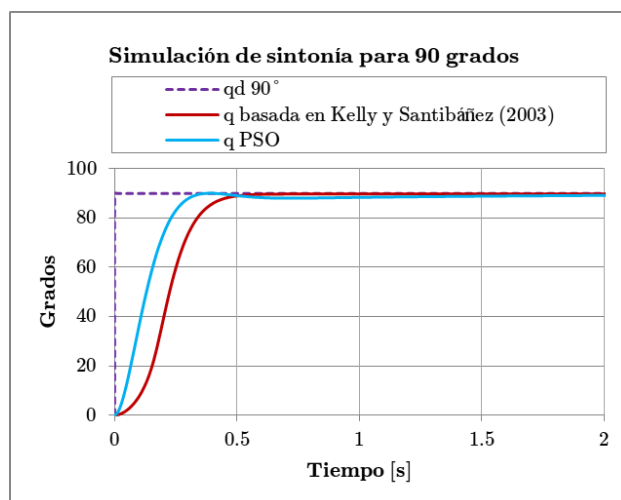


Figura 4. Simulación de sintonía para 90° (2 segundos)

y la figura 8 representan las evaluaciones y la comparativa de las respuestas respectivamente.

Tabla IV
EVALUACIÓN DE LA SIMULACIÓN DE SINTONÍA PARA 90°

Simulación de sintonía para 90°		
Criterios de desempeño del controlador	Método de sintonía	
	Basado en Kelly y Santibáñez (2003)	PSO
Sobre-impulso (grados)	0	0.112
Error en estado estacionario (grados)	0	0
Tiempo de subida (segundos)	0.24	0.2
Tiempo de establecimiento (segundos)	0.45	0.3
Voltaje máximo antes del limitador (volts)	9.31	5.946
Valor máximo del par (Newton metro)	1.174	0.697

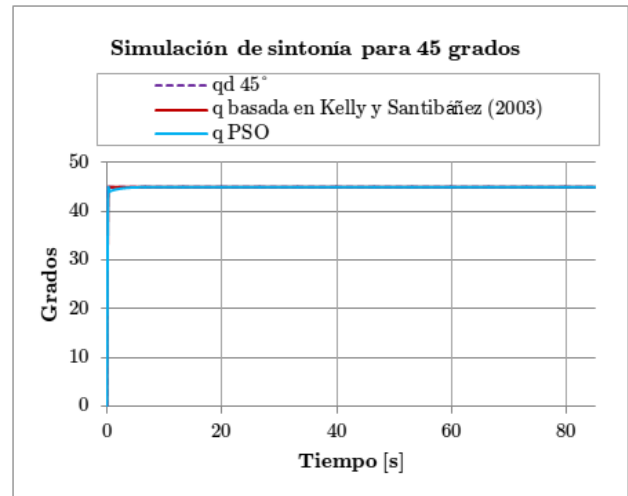


Figura 6. Simulación de sintonía para 45°

Porcentaje de diferencia de sintonía PSO con respecto a sintonía basada en Kelly y Santibáñez (2003) en simulación para 90 grados

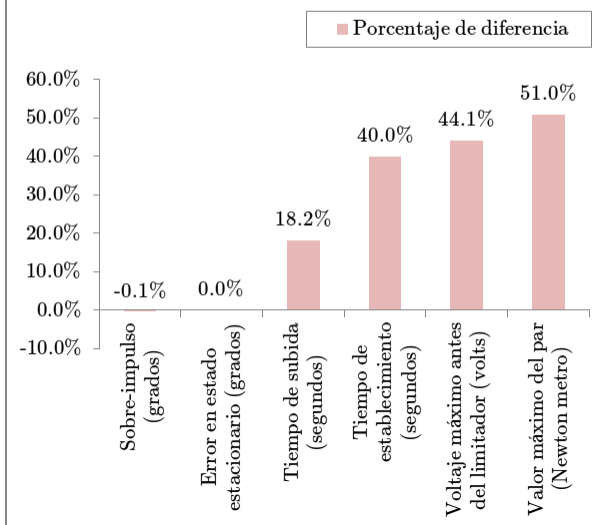


Figura 5. Comparativa de la simulación de sintonía para 90°

Tabla V
EVALUACIÓN DE LA SIMULACIÓN DE SINTONÍA PARA 45°

Simulación de sintonía para 45°		
Criterios de desempeño del controlador	Método de sintonía	
	Basado en Kelly y Santibáñez (2003)	PSO
Sobre-impulso (grados)	0.038	0
Error en estado estacionario (grados)	0	0
Tiempo de subida (segundos)	0.16	0.14
Tiempo de establecimiento (segundos)	0.24	0.2
Voltaje máximo antes del limitador (volts)	6.116	5.892
Valor máximo del par (Newton metro)	0.747	0.717

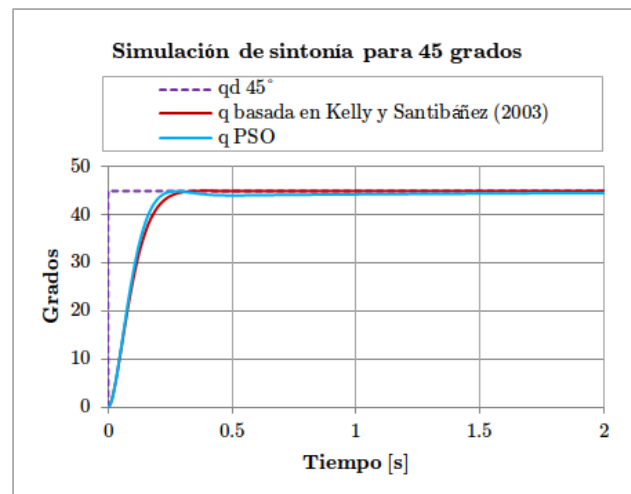


Figura 7. Simulación de sintonía para 45°

Cabe destacar que como se muestra en las tablas IV y V la magnitud de los voltajes máximos medidos utilizando la sintonía PSO no excede el voltaje nominal del equipo (6 [V]) a diferencia de cuando se aplicó la sintonía basada en Kelly y Santibáñez (2003) que si lo excede.

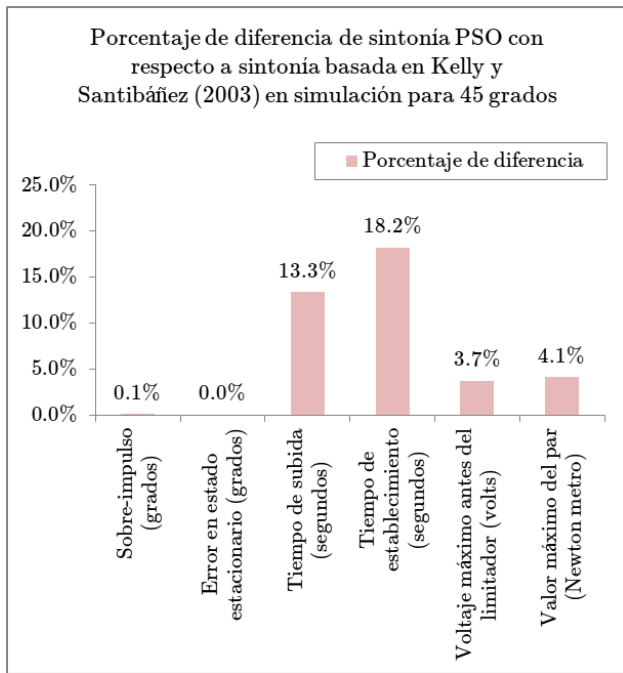


Figura 8. Comparativa de la simulación de sintonía para 45°

Como se puede observar en las comparativas de las figuras 5 y 8 la sintonía PSO obtuvo mejores resultados en la mayoría de los criterios de desempeño excepto en el error en estado estacionario, donde se obtuvo el mismo valor en ambos métodos de sintonía, y el sobre-impulso cuando $q_d = 90^\circ$ donde la sintonía basada en Kelly y Santibañez (2003) obtuvo un mejor resultado en un 0.1%.

IV-B. Evaluación experimental

Para el experimento se utilizó el equipo SRV02 de la marca Quanser, bajo la configuración mostrada en la figura 9 como posición inicial, colocando la barra de carga en su punto medio sobre el eje de carga. Posteriormente se colocó el disco sobre uno de los extremos de la barra de carga, el cual se posicionó sobre la marca de 0° situada en el cuerpo metálico del equipo. La cara del equipo donde sobresale el eje de carga se colocó de manera vertical, quedando situado el disco apuntando hacia el suelo. Para la toma de datos de posición y la velocidad del disco, se utilizó el encoder integrado en el equipo (US Digital S1) [16]. Después de cada ejecución el equipo fue colocado en la posición inicial.

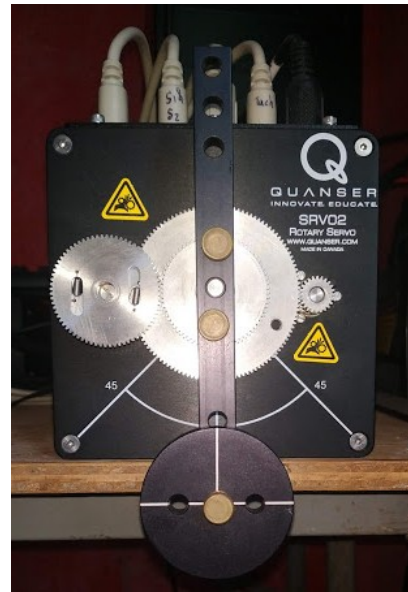
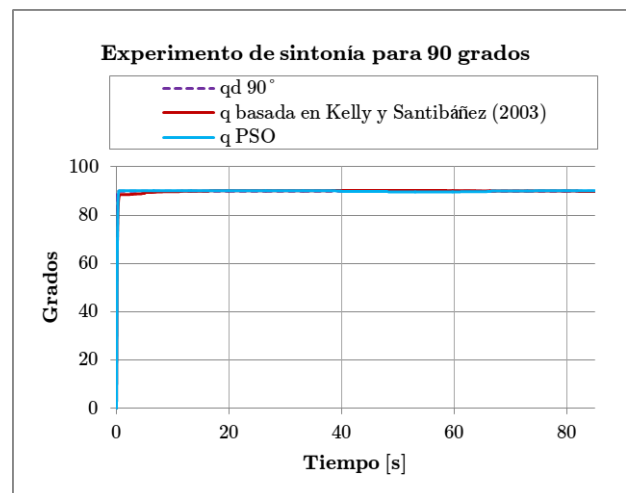


Figura 9. SRV02 de Quanser como articulación motriz pendular

A continuación se muestran los resultados experimentales cuando $q_d = 90^\circ$, donde las figuras 10 y 11 representan las respuestas del sistema, en la primera se observa el registro completo de los 85 segundos de muestreo de q y la segunda es un acercamiento a los primeros 2 segundos del muestreo de q . La tabla VI y la figura 12 representan las evaluaciones y la comparativa de las respuestas de los dos métodos de sintonía respectivamente. De igual manera para el caso donde $q_d = 45^\circ$, las figuras 13 y 14 representan las respuestas del sistema, mientras la tabla VII y la figura 15 representan las evaluaciones y la comparativa de las respuestas respectivamente.


 Figura 10. Experimento de sintonía para 90° (85 segundos)

Cabe destacar que como se muestra en las tablas VI y VII la magnitud de los voltajes máximos medidos utilizando la sintonía PSO no excede el voltaje nominal del equipo (6 [V])

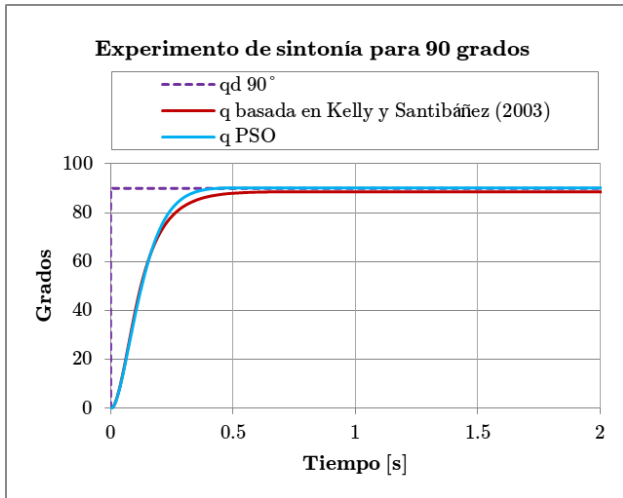


Figura 11. Experimento de sintonía para 90° (2 segundos)

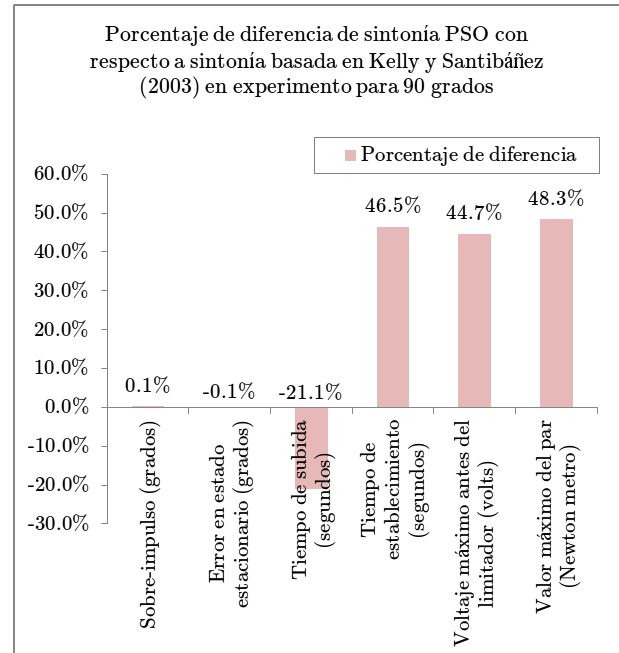


Figura 12. Comparativa del experimento para 90°

Tabla VI
EVALUACIÓN DEL EXPERIMENTO DE SINTONÍA PARA 90°

Experimento de sintonía para 90°		
Criterios de desempeño del controladora	Método de sintonía	
	Basado en Kelly y Santibáñez (2003)	PSO
Sobre-impulso (grados)	0.308	0.176
Error en estado estacionario (grados)	0.110	0.176
Tiempo de subida (segundos)	0.17	0.21
Tiempo de establecimiento (segundos)	0.53	0.33
Voltaje máximo antes del limitador (volts)	9.319	5.913
Valor máximo del par (Newton metro)	1.179	0.72

Tabla VII
EXPERIMENTO DE SINTONÍA PARA 45°

Experimento de sintonía para 45°		
Criterios de desempeño del controlador	Método de sintonía	
	Basado en Kelly y Santibáñez (2003)	PSO
Sobre-impulso (grados)	0.33	0.154
Error en estado estacionario (grados)	0	0.44
Tiempo de subida (segundos)	0.18	0.15
Tiempo de establecimiento (segundos)	0.29	0.28
Voltaje máximo antes del limitador (volts)	6.217	5.948
Valor máximo del par (Newton metro)	0.786	0.752

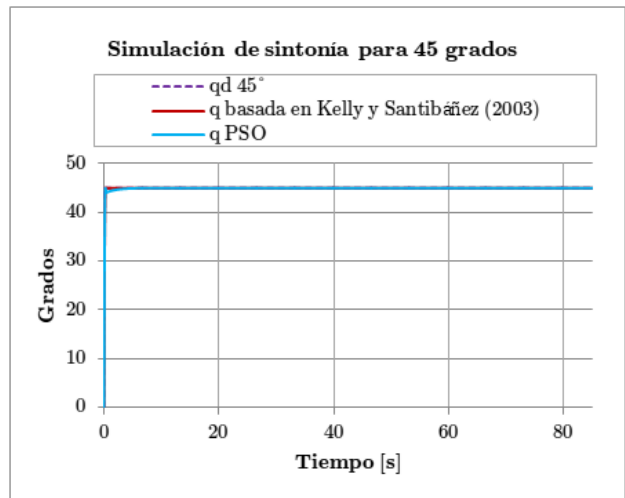


Figura 13. Experimento de sintonía para 45° (85 segundos)

a diferencia de cuando se aplicó la sintonía basada en Kelly y Santibáñez (2003) que si lo excede.

Como se puede observar en las comparativas de las figuras 12 y 15 la sintonía PSO obtuvo mejores resultados en la mayoría de los criterios de desempeño excepto en el error en estado estacionario y el sobre-impulso cuando $q_d = 90^\circ$ donde la sintonía basada en Kelly y Santibáñez (2003) obtuvo un mejor resultado en un 0.1 y 21.1 respectivamente, y en el error en estado estacionario cuando $q_d = 45^\circ$ donde la sintonía basada en Kelly y Santibáñez (2003) obtuvo un mejor resultado en un 0.1.

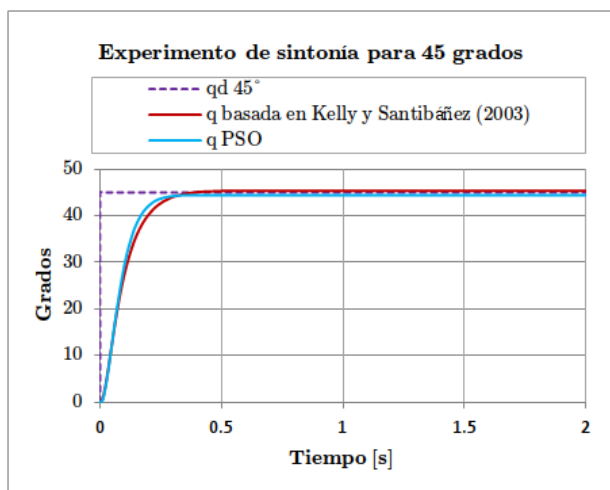


Figura 14. Experimento de sintonía para 45° (2 segundos)

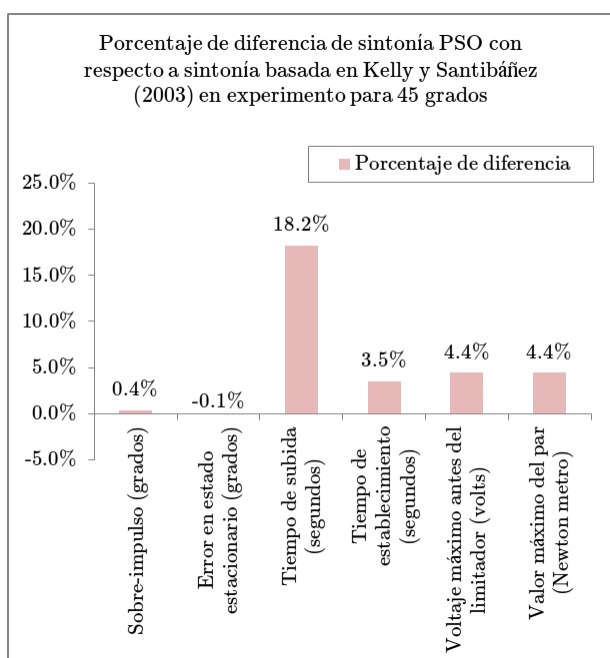


Figura 15. Comparativa del experimento para 45°

V. CONCLUSIONES

En este trabajo se mostró que la estrategia de sintonía mediante PSO con el esquema de aptitud presentado en la subsección III-A, obtuvo mejores resultados en la mayoría de los criterios de desempeño, que la sintonía que utilizaba las propiedades establecidas por Kelly y Santibáñez (2003) [7] en simulaciones y experimentos, como se muestra en las graficas comparativas de las figuras 5, 8, 12 y 15.

La inclusión de la magnitud del voltaje y el voltaje nominal como un índice de desempeño muestra que se puede lograr una sintonía adecuada, que permite a una articulación motriz pendular cumplir con $q_d = 90^\circ$ y $q_d = 45^\circ$ sin la necesidad

de sobrepasar el voltaje nominal.

Queda para futuras investigaciones llevar a cabo la sintonía de controladores PID aplicados en la regulación de posición de robots tipo articulación motriz pendular, con un mayor número de grados de libertad, la implementación de técnicas de optimización heurísticas multi-objetivo que permita realizar la sintonía tomando en cuenta una serie de funciones de aptitud de manera simultánea, que sea capaz de encontrar un conjunto de soluciones no dominadas [8], y comparar la estrategia con otros métodos de sintonía establecidos dentro del área de la ingeniería de control.

VI. AGRADECIMIENTOS

Se agradece al Consejo Nacional de Ciencia y Tecnología (CONACYT) por brindar su apoyo a través de la beca para estudios de maestría y permitir la elaboración a tiempo completo de esta investigación. Así mismo se agradece al Tecnológico Nacional de México (TecNM) y al Instituto Tecnológico de la Paz (ITLP) por facilitar el equipo para las pruebas experimentales.

REFERENCIAS

- [1] M. A. Azar, S. L. Martínez, E. E. Tarifa, S. F. Domínguez, J. J. Gutiérrez, y E. Arte, "Sintonía de Controladores Inteligentes Mediante Estrategia Híbrida Fuzzy-PSO Contexto," *Workshop de Investigadores en Ciencias de la Computación*, pp. 830–836, 2016.
- [2] L. Santos, F. Durand, y T. Abrão, "Sintonía adaptativa de algoritmos PID de controle de potência em redes ng-pon baseado em PSO," in *Anais de XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*. São Pedro: Sociedade Brasileira de Telecomunicações, 2017, pp. 3–6. [Online]. Disponible en: <http://biblioteca.sbrt.org.br/articles/441>
- [3] R. Valerdi, "Heuristics for systems engineering cost estimation," *IEEE Systems Journal*, vol. 5, pp. 91–98, mar 2011. [Online]. Disponible en: <http://ieeexplore.ieee.org/document/5556044/>
- [4] C. Blum y A. Roli, "Metaheuristics in combinatorial optimization," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, sep 2003. [Online]. Disponible en: <https://dl.acm.org/doi/10.1145/937503.937505>
- [5] J. Kennedy y R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4. Perth: IEEE, 1995, pp. 1942–1948. [Online]. Disponible en: <http://ieeexplore.ieee.org/document/488968/>
- [6] O. Arrieta Orozco y V. M. Alfaro Ruíz, "Sintonización de controladores PI y PID utilizando los criterios integrales IAE e ITAE," *Ingeniería Revista de la Universidad de Costa Rica*, vol. 13, no. 1-2, jul 2011. [Online]. Disponible en: <http://revistas.ucr.ac.cr/index.php/ingenieria/article/view/618>
- [7] R. Kelly y V. Santibáñez, *Control de Movimiento de Robots Manipuladores*, Madrid, 2003.
- [8] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, vol. 3. Washington: IEEE, 1999, pp. 1951–1957. [Online]. Disponible en: <http://ieeexplore.ieee.org/document/785513/>
- [9] J. A. Morales Viscaya, A. Rochín Ramírez, M. A. Castro Liera, y J. A. Sandoval Galarza, "AG y PSO como métodos de sintonía de un PID para el control de velocidad de un motor CD," *Nuevos avances en robótica y computación*, pp. 81–87, 2015. [Online]. Disponible en: <http://posgrado.itlp.edu.mx/CIRC2015/CIRC2015.pdf>
- [10] S. GirirajKumar, D. Jayaraj, y A. Kishan, "PSO Based Tuning of a PID Controller for a High Performance Drilling Machine," *International Journal of Computer Applications*, vol. 1, no. 19, pp. 12–18, 2010.
- [11] L. H. S. De Andrade, B. L. G. Costa, y B. A. Angélico, "PSO aplicado a sintonia do controlador PI/PID da malha de nível de uma planta didática industrial." Fortaleza: Universidade Tecnológica Federal do Paraná, 2013, p. 6. [Online]. Disponible en: <https://docplayer.com.br/6043150-Pso-aplicado-a-sintonia-do-controlador-pi-pid-da-malha-de-nivel-de-uma-planta-didatica-industrial.html>

- [12] Z. Jun y Z. Kanyu, "A particle swarm optimization approach for optimal design of PID controller for temperature control in HVAC," *Proceedings - 3rd International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2011*, vol. 1, no. 2, pp. 230–233, 2011.
- [13] T. T. Tay, I. M. Y. Mareels, y J. B. Moore, *High Performance Control*, ser. Systems and Control: Foundations and Applications. Boston: Birkhauser, 1997.
- [14] A. B. d. S. Serapião, "Busca harmônica aplicada à sintonia automática de controladores pid a," *Simpósio Brasileiro de Automação Inteligente (SBAI)*, vol. X, pp. 99–104, 2011. [Online]. Disponible en: <https://fei.edu.br/sbai/SBAI2011/81013.pdf>
- [15] A. Oi, C. Nakazawa, T. Matsui, H. Fujiwara, K. Matsumoto, H. Nishida, J. Ando, y M. Kawaura, "Development of PSO-based PID tuning method," in *2008 International Conference on Control, Automation and Systems, ICCAS 2008*. Tokyo: IEEE, oct 2008, pp. 1917–1920. [Online]. Disponible en: <http://ieeexplore.ieee.org/document/4694410/>
- [16] QUANSER, "Rotatory motion servo plant: SRV02 user manual," p. 31, 2008. [Online]. Disponible en: <http://www2.hawaii.edu/~gurdal/EE351L/srv02.pdf>

Estado actual de los coches eléctricos en México, 2021

Alejandro-Israel Barranco-Gutiérrez
Cátedras CONACyT- TecNM en Celaya, México
israel.barranco@itcelaya.edu.mx

Francisco J. Pérez-Pinal
TecNM en Celaya, México
francisco.perez@itcelaya.edu.mx

José-Alfredo Padilla-Medina
TecNM en Celaya, México
alfredo.padilla@itcelaya.edu.mx

Juan-José Martínez-Nolasco
TecNM en Celaya, México
juan.martinez@itcelaya.edu.mx

Resumen— En este trabajo se presenta una revisión al estado actual de los coches comerciales en México, desde el punto de vista de su arquitectura y sus capacidades. Esto con la finalidad de analizar las posibilidades de mejorar las especificaciones de los coches eléctricos para que en un futuro se puedan proponer diseños realizados por ingenieros mexicanos. Los datos son totalmente documentales y con un rango de error por medir, pero sirve como una aproximación que resulta útil para realizar las pruebas respectivas. Se analiza de igual manera la autonomía de estos con carga máxima y su eficiencia energética.

Palabras clave— Coches eléctricos, eficiencia energética, autonomía.

I. INTRODUCCIÓN

El cambio climático y el calentamiento global son temas discutidos rigurosamente por muchos gobiernos desde principios del siglo XXI. Un gran número de informes notables han revelado el impacto negativo del cambio climático, impulsado predominantemente por actividades humanas. Con el aumento de la civilización y la industrialización a nivel mundial, una gran cantidad de quema de combustibles fósiles en las industrias ha llevado al grave problema de la contaminación del aire [1]. Al mismo tiempo, no se pueden ignorar las emisiones de escape de los vehículos automotores. Las emisiones de vehículos, que incluyen principalmente bióxido de carbono (CO₂), monóxido de carbono (CO), óxidos de nitrógeno (NO_x) y partículas suspendidas (PM10 y PM2.5), han sido consideradas como los principales contribuyentes de los gases de efecto invernadero, lo que también ha llevado al aumento de diferentes formas de cáncer y otras enfermedades graves [2][3]. El sector del transporte en constante crecimiento consume alrededor del 49% de los recursos petroleros. Siguiendo las tendencias actuales del consumo de petróleo y las fuentes de petróleo crudo, se prevé que los recursos de petróleo del mundo se agoten para 2038 [4]. Por tanto, la sustitución de los recursos energéticos no renovables por fuentes de energía renovables, y el uso de tecnologías adecuadas de ahorro energético parece ser obligatorio. Los coches eléctricos (CE) como una posible solución para aliviar los problemas ambientales relacionados con el tráfico, se han investigado y estudiado ampliamente [5-7]. En comparación con coches de combustión interna, las características atractivas de los CE son principalmente la fuente de alimentación y el sistema de propulsión.

México enfrenta tres grandes problemas en materia energética. El primero es la inminente escasez de los combustibles fósiles, así como la alta dependencia de sus derivados refinados y gas natural, mismos que son importados. Por lo cual la seguridad energética del país es frágil. El segundo es el impacto ambiental que conlleva la producción y consumo de energía tal como, el uso y contaminación del agua, contaminación de suelos y aire. El tercero es la inequidad energética pues más de 40 millones de familias en México en localidades rurales vive en pobreza energética [8]. Uno de los sectores que más energía consume es el de transporte. El balance nacional (mexicano) de energía de 2017, reportó que el autotransporte utiliza el 90% de la energía en este sector como se muestra en Figura 1.

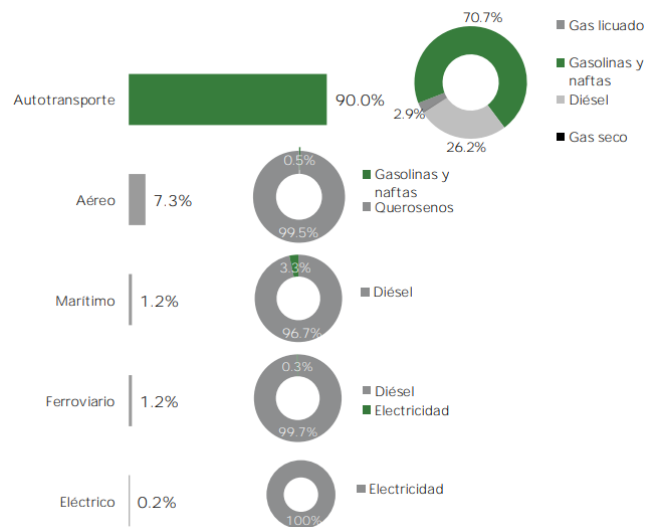


Fig. 1. Consumo de energía del sector transporte, 2017 (Estructura porcentual por subsector y energético)

El hecho de que el autotransporte consuma tales cantidades de energía obliga a los mexicanos a cambiar los hábitos actuales de transportación y consumo de energía. Una de las razones fundamentales por las cuales los CE han cobrado relevancia es porque distintas fuentes de energía renovable pueden convertirse en energía eléctrica, y esta puede almacenarse en baterías y transportarse por cables o mecánicamente. Este hecho predice

que pronto en México se transitará al uso de CE, como varios autores y fabricantes lo han estimado. Por lo cual es de vital importancia revisar el estado actual de esta tecnología que permite generar negocio y diseños mexicanos. Teniendo en cuenta las condiciones ambientales y la disponibilidad reducida de combustible, los CE's desempeñan un papel vital en muchas aplicaciones, como los automóviles de consumo y el transporte de corta distancia. En [27] proponen un modelo dinámico detallado de batería, motor e inversor desarrollado para el diseño de un CE. Además, se desarrolla un controlador mejorado con un método geométrico diferente utilizando la ganancia de sensibilidad del sensor de corriente y el tacómetro para asegurar el rendimiento óptimo del CE. Para lograr un funcionamiento lineal del vehículo y una estabilidad mejorada, se diseña un modelo de función de transferencia del sistema considerando diversas incertidumbres, como la fuerza que actúa sobre el automóvil, la rueda, la carretera y las condiciones de velocidad del viento. Usan el software MATLAB / Simulink, para analizar su rendimiento a través de varios escenarios de prueba de bucle abierto y de bucle cerrado. En [28], los autores proponen un marco de control de cruce adaptativo que considera el frenado regenerativo para mejorar la seguridad y la eficiencia energética de los CE durante el proceso de seguimiento del automóvil. Al principio, se construye un modelo dinámico acoplado y no lineal del sistema CE, que se compone principalmente de una batería potente, un motor eléctrico, una transmisión de una sola velocidad y un sistema de frenado hidráulico. Luego, se diseña un controlador de alto nivel de modo deslizante difuso adaptativo para obtener con precisión la aceleración longitudinal deseada de los CE, en el que se utiliza la lógica difusa para aproximar el elemento de control de conmutación del control del modo deslizante para la vibración libre. Y la estabilidad del controlador de alto nivel está probada por la teoría de Lyapunov. En el controlador de nivel inferior, el control de tracción y el control de frenos están diseñados para rastrear la aceleración deseada producida por el controlador de alto nivel, además, se presenta una nueva estrategia de frenado regenerativo para maximizar la recuperación de energía de frenado. Finalmente, los resultados de la simulación indican que el esquema de control propuesto tiene un excelente desempeño de seguimiento longitudinal y recuperación de energía de frenado sin pérdida de seguridad. El trabajo propuesto por Mahdiraji et al, optimizan la combinación de generación distribuida y vehículos eléctricos mediante el uso de métodos probabilísticos para mejorar la confiabilidad de la microrred. Optimizar la combinación de generación de energía con vehículos eléctricos para mejorar la confiabilidad de las microrredes, proporcionar una nueva forma de planificar la carga y descarga de los vehículos y modelar la inyección de energía en las microrredes son algunas de las innovaciones de este documento. También en [30] los autores hacen una revisión completa de diferentes tipos de tecnologías de extensión de rango de vehículos eléctricos, incluidos motores de combustión interna, generadores lineales de pistón libre, celdas de combustible, micro turbinas de gas y baterías de zinc-aire, y describe sus definiciones, mecanismos de trabajo y algunos desarrollos recientes. de cada gama ampliando la tecnología. También se presenta una comparación entre las diferentes tecnologías, destacando las ventajas y desventajas de cada una, para ayudar a abordar las necesidades de investigación futuras. Dado que los vehículos eléctricos serán una parte

importante del futuro de la industria automotriz, los extensores de rango serán un concepto importante a explorar para proporcionar una solución rentable, confiable, eficiente y dinámica para combatir el problema de ansiedad de rango que tienen los consumidores actualmente.

II. ARQUITECTURA DEL COCHE ELÉCTRICO

“El término carro proviene del pueblo de Kocs, donde, en 1518, se empezaron a construir carruajes especiales para transportar gente de un lugar a otro. Así se formó la palabra kocsí /kóchi/, que los españoles empezaron a utilizar como coche en escritos a partir de 1548, para referirse al carro cubierto. Es decir, el coche es aquel vehículo que transporta personas” [10]. Por lo que, el término CE lo concebimos como un vehículo que transporta personas utilizando energía eléctrica. Los CE, en comparación con los vehículos clásicos con motor de combustión interna, son más simples y fáciles de operar. La arquitectura de tren motriz más simple consiste en una batería de bajo o alto voltaje, un motor eléctrico (asíncrono, síncrono o de corriente directa) con controlador de electrónica de potencia (corriente directa o alterna) y una caja de cambios de una sola velocidad. Es necesario mencionar que, debido a necesidades de reemplazar el uso de imanes de tierras raras, actualmente existe un gran esfuerzo de la comunidad científica en la investigación, desarrollo e innovación acerca de los motores de reluctancia variable [20].

Los CE también se denominan vehículos eléctricos puros, para distinguirlos de los vehículos eléctricos híbridos, que tienen un tren de potencia híbrido (motor de combustión interna más motor eléctrico). Uno de los componentes centrales del CE es el motor eléctrico junto con su sistema de enfriamiento, sensores y caja de velocidades como se ilustra en la Figura 2. El chasis es la estructura metálica del automóvil que soporta a las demás partes y sistemas, además de alojar y proteger a los pasajeros. Se conforma por elementos rígidos y fuertes para resistir todos los golpes ocasionados por las irregularidades del camino, la carrocería y posibles colisiones con cuerpos externos. En la generación actual de CE se utiliza el chasis tipo plataforma independiente que aloja también el sistema de baterías, como se muestra en la Figura 3.



Fig. 2. Fotografía de un motor de un coche Tesla junto con su caja de engranes, sistema de enfriamiento y sensores.



Fig. 3. Chasis de coche Tesla armado con llantas, amortiguadores, sistema mecánico de dirección y baterías.

Otro elemento importante es la carrocería, que es la capa exterior de un vehículo, lo que permite dar forma y utilidad a un chasis o plataforma. Esta es la estructura básica que permite crear el habitáculo en el que se situarán los pasajeros y la carga. Además, es la que permite dotar de una estética y funcionalidad concretas al mismo, pues en la actualidad muchos modelos diferentes parten del mismo chasis o plataforma como muestra la imagen de la Figura 4, donde se aprecia la carrocería del coche Twizy de Renault.

De igual manera, otro componente central en los CE es la unidad de almacenamiento de energía. Para esta puede utilizarse al menos una combinación de dos de los siguientes elementos: baterías, celdas de combustible, volantes inerciales, o supercapacitores. Los tipos de batería que suelen utilizarse en el CE son generalmente de plomo ácido, iones de litio (Li-Ion), de hidruro metálico de níquel (Ni-MH), ZEBRA (Na-NiCl₂) y azufre de litio (Li-S) [21]. La batería, que es lo más pesado en cualquier CE, se instala generalmente debajo de los asientos, colocando el centro de gravedad tan bajo como es posible e incrementando la rigidez estructural y estabilidad comparada con otro coche convencional.



Fig. 4. Carrocería de coche Twizy de dos plazas.



Fig. 5. Conjunto de baterías del coche eléctrico Leaf de Nissan.

Actualmente, se tiene un estimado que el paquete de baterías retendrá entre el 70% y el 80% de su capacidad después de 10 años, pero su longevidad depende de la frecuencia de las recargas rápidas y de otros factores ambientales como el tiempo caluroso extremo.

De acuerdo con la norma SAE J1772, actualmente existen tres niveles para la recarga de baterías [22]. El cargador de nivel 1 simplemente se carga desde un tomacorriente doméstico estándar de 120V. Un cargador de nivel 2 requiere esencialmente un suministro de voltaje superior a 200 voltios. En contraste, un cargador de nivel 3 utiliza esencialmente una corriente continua en un voltaje mucho más alto y puede cargar algunos vehículos eléctricos enchufables con hasta 800 voltios. Si la carga rápida es el método habitual de carga se producirá una pérdida de capacidad adicional del 10% en 10 años, comparado con la carga habitual de 220 V. La capacidad quedaría en el 70% tras 10 años de recargas rápidas como método habitual de recarga. En la Figura 5 se aprecia el conjunto de baterías del coche eléctrico de Nissan.

El flujo de corriente en las baterías de los coches eléctricos provoca el calentamiento de sus celdas. Cuanto mayor es el flujo de la corriente, mayor es el aumento de temperatura. El rendimiento de las baterías depende de su temperatura. Llamado efecto Goldilocks, es decir, las baterías no funcionan adecuadamente cuando están sometidas a temperaturas extremas. Esto puede provocar daños permanentes en las baterías y una degradación acelerada. Por lo cual, se requiere de un sistema de control de temperatura para las baterías, que las enfríen o calienten cuando sea necesario. La Figura 6 muestra el sistema de control de temperatura para las baterías del BMW i3.

Es importante mencionar que una opción que están buscando los fabricantes de CE es que la recarga de baterías sea totalmente independiente. Para esto se han realizado esfuerzos en desarrollar sistemas de transferencia de potencia inalámbrica (TPI). La TPI se puede realizar mediante varios medios como son: el acústico, luminoso, microondas, capacitivo e inductivo. Los sistemas funcionan al estacionarse un CE sobre una plataforma de montaje en tierra compatible con SAE J2954.



Fig. 6. Conjunto de baterías y sistema de control de temperatura en anaranjado del BMW i3

Para este fin, la energía se transfiere creando un campo de resonancia magnética entre la plataforma de transmisión en el suelo y una plataforma de recepción instalada en la parte inferior del CE. La energía atraviesa un espacio de aire y luego se convierte de corriente alterna a corriente directa en el CE. La SAE J2954 establece tres niveles de potencia: WPT1 (3,7 kW), WPT2 (7 kW) y WPT3 (11 kW) [23].

Finalmente es necesario mencionar que existen a nivel mundial diferentes configuraciones del tren motriz de los CE. La primera configuración contiene un motor eléctrico, un embrague, una caja de cambios y un diferencial mecánico. La segunda contiene un motor eléctrico único con engranaje fijo. La tercera esta basada en un motor eléctrico con engranes fijos y diferencial integrados en un solo empaque. La cuarta

implementa un diferencial eléctrico, el cual es implementado con dos motores eléctricos en las ruedas. La quinta configuración contiene un sistema de tracción en las ruedas (in-wheel). La última configuración contiene un motor eléctrico de rotor exterior de baja velocidad dentro de las ruedas. Cada una de ellas presentan diferentes ventajas y limitaciones, para un mayor detalle vea [24].

III. CARACTERÍSTICAS DE COCHES ELÉCTRICOS DISPONIBLES EN MÉXICO

La información recuperada de los diferentes coches eléctricos, fue obtenida de los sitios web mexicanos de las respectivas marcas. Entre la información más importante destacan la autonomía, el tiempo de carga de las baterías y el peso total del coche. En las Tablas 1a y 1b podemos comparar tales características. Es necesario mencionar que algunas marcas no proporcionan públicamente toda la información técnica requerida para realizar una comparación más exhaustiva.

Analizando la Tabla 1a, puede observarse que existen seis grandes empresas que comercializan sus CE. Donde la empresa Nissan oferta dos versiones de su CE Leaf, siendo la diferencia entre ellos su batería. Por el contrario, el modelo Bolt tiene una batería casi del doble de capacidad comparado al Leaf, lo cual se refleja en la duplicidad de su tiempo de autonomía. De igual manera la empresa Tesla oferta tres tipos de CE. Donde la capacidad instalada de baterías, peso y potencia generada son sus principales diferencias.

TABLA 1A. CARACTERÍSTICAS PRINCIPALES DE VEHÍCULOS ELÉCTRICOS DISPONIBLES EN MÉXICO: NISSAN, CHEVROLET, RENAULT Y BMW.

Marca	Nissan		Chevrolet	Renault	BMW
Nombre	Leaf 24 Kw eléctrico 17	Leaf 30 Kw eléctrico 17	BOLT EV	Twizy	i3 (94Ah)
Batería	Ion – litio laminada 24 kWh, 192 celdas, 360V	Ion – litio laminada 30 kWh, 192 celdas, 360V	Ion – litio 60 kWh	Ion – litio 6.1 kWh	Iones de litio 27.2
Motor	CA síncrono de corriente alterna	CA síncrono de corriente alterna	CA síncrono de corriente alterna	3CG eléctrico síncrono con rotor bobinado	
Potencia	152HP	220HP	200 HP	20 HP	184HP
Capacidad nominal					103 HP / 48 rpm
Peso del auto (Kg)	1509	1544	1,616	474	1320
Peso bruto vehicular	1901 kg	2010 kg		Carga útil 110 kg máximo 126 kg	1365 kg
Torque	254 Nm / 187(lb - ft)	254 Nm/ 187(lb - ft)	360 Nm / 265.5 (lb - ft)	57 Nm / 42 (lb - ft)	270 Nm / 200 (lb - ft)
Dimensiones exteriores:					
Alto (mm)	1550	1550	1594	1454 (1980 c/ puerta abierta)	1578
Ancho (mm)	1770	1770	1765	1234	2039
Largo (mm)	4445	4445	4166	2338	3999
Tiempo de recarga	100% en 4 horas a 220V	100% en 4 horas a 220V			
Autonomía (con carga completa)	160 km	160 km	383 km	100 km	250 km

Consumo eléctrico	19,4 kWh/100 km	19,4 kWh/100 km			12.9 kW/h cada 100 km
--------------------------	-----------------	-----------------	--	--	-----------------------

TABLA 1B. CARACTERÍSTICAS PRINCIPALES DE VEHÍCULOS ELÉCTRICOS DISPONIBLES EN MÉXICO: TESLA Y JAGUAR.

Marca	Tesla			Jaguar
Nombre	Modelo 3	Modelo S	Modelo X	I-PACE
Batería	50-75 kWh 360 V ion-litio	100 kWh 350 V ion-litio	100 kWh 350 V ion-litio	90 kWh 388 V ion-litio
Motor	Un motor: 258 HP (262 CV). Versión dual motor: 154 kW (209 CV) y 197 kW (268 CV).	AC 302 HP	Motor delantero de 262 HP y motor trasero de 510 HP. La potencia máxima del conjunto es de 610 HP	Dos motores que combinados logran 294 kW
Potencia	301 a 483 HP	301 HP	200 HP	395 HP
Capacidad nominal	350 HP			
Peso del auto (Kg)	1509	2108	2459	2208
Peso bruto vehicular	2060 kg	2108 kg	2459 kg	2670 kg
Torque	420 a 660 N·m	755 a 1140 N·m	755 a 1140 N·m	696 N·m
Dimensiones exteriores:				
Alto (mm)	1443	1445	1684	1565
Ancho (mm)	1933	1964	1999	1895
Largo (mm)	4694	4970	5052	4682
Tiempo de recarga	100% en 4 horas a 220V	100% de 4 a 10 horas		80% Diez horas
Autonomía (con carga completa)	360 km	600 km	525 km	415 km
Consumo eléctrico	15.5 kW/h cada 100 km	16.4 kWh /100 km	26 kWh a los 100 km	26 kWh a los 100 km

Analizando ambas tablas, puede observarse que el CE con menor potencia es el Twizy, el de mayor el modelo 3, en donde este último tiene una potencia superior de hasta 20 veces. Esta característica entre ambos CE tiene una relación directa con el peso del auto, el modelo 3 tiene un peso bruto de 2060kg, en contraste con los 474kg del Twizy. En general el Twizy se considera un CE ligero de vecindario y de autonomía limitada. El cual lo hace una buena opción en las grandes urbes, donde la velocidad promedio es de 25km/h [25]. En contraste el modelo 3 esta diseñado para grandes distancias entre puntos de recarga.

Actualmente existen más de 200 puntos de recarga públicos en el país, también conocidos como electrolineras [26]. Estas se ubican en lugares de acceso público como centros comerciales, tiendas de conveniencia, y gasolineras comerciales. De acuerdo con datos del Programa de Ahorro de Energía del Sector Eléctrico (PAESE), que es el área que monitorea las electrolineras en el País, hay 893 puntos de recarga con un total de 2 mil 17 electrolineras. Donde la CDMX, Nuevo León y Jalisco son las entidades con el mayor número de estas con un total de 384, 180 y 177, respectivamente.

IV. DISCUSIÓN

Dos parámetros que permiten mejorar el diseño del coche eléctrico son la eficiencia energética y la autonomía del vehículo. Algunos fabricantes no presentan esta información en

sus sitios web, por lo que esa información no se tiene en las gráficas siguientes. También es importante observar que son estimaciones que dependen de las condiciones de manejo, así como condiciones donde se desarrolla el mismo. Por lo cual es un comparativo documental desde internet [14-19].

En la Figura 7, podemos observar como el i3 cuenta con muy buena eficiencia energética ya que es el que menor consumo eléctrico tiene comparado con los otros seis. Así mismo, el modelo X, y el I-PACE tienen un consumo eléctrico similar. Sin embargo, en cuestiones de potencia el I-PACE duplica al modelo X. Finalmente en el rango intermedio de consumo esta el modelo 3 y S, los cuales consumen un poco más de 15 kW/hr por cada 100 km.

Analizando la Figura 8 se logra observar como el Modelo S cuenta con la mayor autonomía de todos los CE comparados. De hecho, duplica la autonomía del modelo i3, y triplica la del Leaf. Esta gran diferencia en autonomía es debida principalmente a la cantidad y tipo de baterías, peso, potencia generada y estrategia de manejo de energía y/o potencia utilizada. Este último factor también es determinante en el diseño de los CE. Por ejemplo, la estrategia de energía se diseña con el propósito de lograr una mayor autonomía del CE entre carga y recarga. En contraste la estrategia de potencia, generalmente se diseña con la finalidad de mejorar las prestaciones de desempeño i.e. una alta aceleración en reducido tiempo, lograr un par instantáneo. Por lo que es importante tener un balance entre estas estrategias para el adecuado control de un CE. De igual manera, profundizando

en la Fig. 8, se observa que el modelo IPACE tiene una autonomía media, a pesar de su mayor peso.

V. CONCLUSIÓN

En este trabajo se presentó una revisión al estado actual de los coches comerciales en México, desde el punto de vista de su arquitectura y sus capacidades. Se consideraron los CE de las principales empresas distribuidoras, con datos públicos y disponibles al usuario final.

Un aspecto importante a tener en cuenta es el costo de adquisición, mismo que no se determina en este documento. Así como tampoco se consideró la vida útil de las baterías, los gastos de mantenimiento, y los incentivos fiscales que algunas entidades dan para incrementar el uso de este tipo de CE. La finalidad de este trabajo es presentar una revisión del estado actual de los coches eléctricos en México, con el propósito de ser una referencia y observar la evolución de esta tecnología en el país.

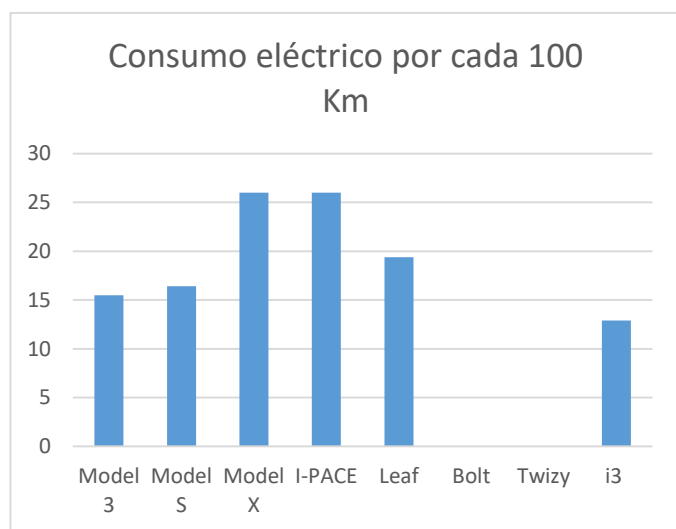


Fig. 7. Eficiencia energética de los modelos investigados por cada 100 km.

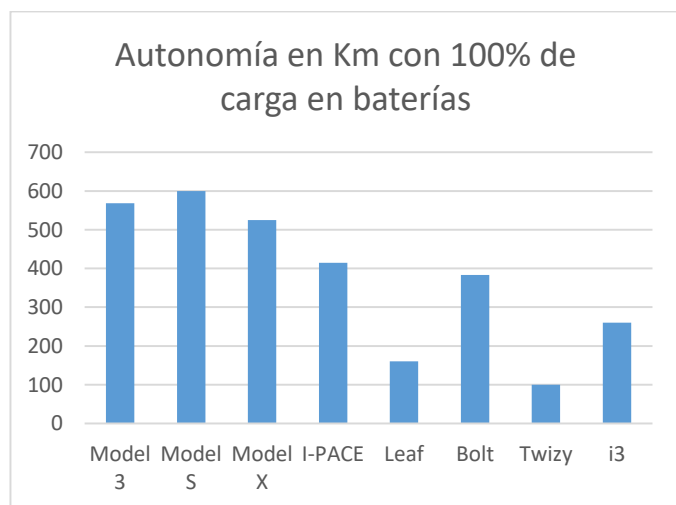


Fig. 8. Autonomía máxima de los modelos investigados con carga máxima.

RECONOCIMIENTOS

Al Consejo Nacional de Ciencia y Tecnología, Tecnológico Nacional de México en Celaya, a La Secretaría de Ciencia y Tecnología de Guanajuato y a Tecnoindustrias.

REFERENCIAS

- [1] Wee, J.H.: Contribution of fuel cell systems to CO2 emission reduction in their application fields. *Renewable and Sustainable Energy Reviews* 14(2), 735–744 (2010).
- [2] Fenton, J. and Hodkinson, R.: *Lightweight Electric/Hybrid Vehicle Design*. 1st. edition, Elsevier, Butterworth-Heinemann (2001).
- [3] Fajri, P. and Asaei, B.: Plug-in hybrid conversion of a series hybrid electric vehicle and simulation comparison, 11th International Conference on Optimization of Electrical and Electronic Equipment, 2008. OPTIM 2008, IEEE, pp.287–292, Brasov (2008).
- [4] Ehsani, M., Gao, Y. and Emadi, A.: *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory, and Design*, 1st edition CRC Press, Boca Raton. (2010)
- [5] Clement, K., Haesen, E. and Driesen, J.: Coordinated charging of multiple plug-in hybrid electric vehicles in residential distribution grids, *Power Systems Conference and Exposition*, 2009. PSCE'09, pp.1–7, IEEE/PES, Seattle, USA, (2009).
- [6] Hajimiragha, A., Canzares, C.A., Fowler, M.W. and Elkamel, A.: Optimal transition to plug-in hybrid electric vehicles in Ontario, Canada, considering the electricity-grid limitations, *IEEE Trans. Ind. Electron*, 57(2), pp.690–701 (2010).
- [7] Stephan, C.H. and Sullivan, J.: Environmental and energy implications of plug-in hybridelectric vehicles, *Environ. Sci. Technol.*, 42 (4), pp.1185–1190. (2008).
- [8] CONVOCATORIA 2020 PARA LA ELABORACIÓN DE PROPUESTAS DE PROYECTOS DE INVESTIGACIÓN E INCIDENCIA PARA TRANSITAR A UN SISTEMA ENERGÉTICO SOCIAL Y AMBIENTALMENTE SUSTENTABLE, <https://www.conacyt.mx/index.php/el-conacyt/convocatorias-y-resultados-conacyt/convocatorias-fordecyt/convocatorias-abiertas-fordecyt/fordecyt-2020-01-energia/20236-demanda-fordecyt-pronaces-2020-01-energia/file>, última consulta 2020/11/11.
- [9] Balance nacional (mexicano) de energía 2017, https://www.gob.mx/cms/uploads/attachment/file/414843/Balance_Nacional_de_Energ_a_2017.pdf, última consulta 2020/11/11.
- [10] Para hablar bien ¿coche, carro o auto?, <https://www.excelsior.com.mx/blog/algarabia/para-hablar-bien-coche-carro-o-auto/1015127>, última consulta 2020/11/11.
- [11] Anatomía de un coche eléctrico, [https://x-engineer.org/automotive-engineering/vehicle/electric-vehicles/anatomy-of-a-battery-electric-vehicle-bev/#:~:text=Battery%20Electric%20Vehicles%20\(BEVs\)%2C,and%20a%20single%20speed%20gearbox](https://x-engineer.org/automotive-engineering/vehicle/electric-vehicles/anatomy-of-a-battery-electric-vehicle-bev/#:~:text=Battery%20Electric%20Vehicles%20(BEVs)%2C,and%20a%20single%20speed%20gearbox), última consulta 2020/11/11.
- [12] Motor eléctrico de un coche tesla, <https://www.quora.com/What-does-a-Tesla-cars-electric-motor-look-like>, última consulta 2020/11/11.
- [13] Chasis de un coche Tesla, <https://imgur.com/gallery/EiNtV>, última consulta 2020/11/11.
- [14] Tesla, https://www.tesla.com/es_mx, última consulta 2020/11/11.
- [15] Jaguar, <https://www.jaguarmexico.com.mx>, última consulta 2020/11/11.
- [16] Nissan, <https://www.nissan.com.mx>, última consulta 2020/11/11.
- [17] Chevrolet, www.chevrolet.com.mx, última consulta 2020/11/11.
- [18] Renault, www.renault.com.mx, última consulta 2020/11/11.
- [19] BMW, www.bmw.com.mx, última consulta 2020/11/11.
- [20] J. D. Widmer, R. Martin, and M. Kimiabeigi, Electric vehicle traction motors without rare earth magnets, *Sustainable Materials and Technologies*, vol. 3, 2015, pp. 7-13.

- [21] C Iclodean *et al*, Comparison of different battery types for electric vehicles, 2017 *IOP Conf. Ser.: Mater. Sci. Eng.* **252** 012058
- [22] https://www.sae.org/standards/content/j1772_201710/ , última consulta 2021/02/24.
- [23] https://www.sae.org/standards/content/j2954_202010/ , , última consulta 2021/02/24.
- [24] A. Karki, S. Phuyal, D. Tuladhar, S. Basnet, and B. P. Shrestha, "Status of pure electric vehicle power train technology and future prospects" *Appl. Syst. Innov.* vol. 3, no. 3: 35.
- [25] <https://www.elfinanciero.com.mx/economia/en-que-ciudad-del-pais-circulas-mas-lentamente-no-es-el-df> , última consulta 2021/02/24.
- [26] <https://www.elfinanciero.com.mx/monterrey/tiene-ml-segundo-lugar-de-electrolineras-en-mexico> 2021/02/24.
- [27] Buddhadeva, S., Sangram, K. R., Pravat, K. R., "Advanced Speed - and - current control approach for dynamic electric car modelling", *IET Electrical Systems in Transportation*, 2021;1-18. <https://doi.org/10.1049/>
- [28] J. Guo, W. Li, J. Wang, Y. Luo and K. Li, "Safe and Energy-Efficient Car-Following Control Strategy for Intelligent Electric Vehicles Considering Regenerative Braking," in *IEEE Transactions on Intelligent Transportation Systems*, 2021. doi: 10.1109/TITS.2021.3066611.
- [29] E. A. Mahdiraji, M. S. Amiri, "Optimization of Electric Vehicles Along with Power Generation Units to Improve Microgrid Reliability", *Quantum Journal Of Engineering, Science And Technology* 2(2): 1-15, 2021.
- [30] T. Manh-Kien, B. Asad, V. Reid, W. Derek, P. Satyam, F. Michael, F. Roydon, "A Review of Range Extenders in Battery Electric Vehicles: Current Progress and Future Perspectives", *World electric vehicle journal*, 2021, 12, 54. <https://doi.org/10.3390/wevj12020054>

Máquina de Vectores de Soporte para la Detección de Datos Anómalos en Dispositivos IoT

Jesus A. Lechuga O., Clara Cruz R., Rogelio Reyes R.

Instituto Politécnico Nacional, ESIME Unidad Culhuacán

Av. Santa Ana No. 1000, Col. San Francisco Culhuacán, CP. 04430, México, Ciudad de México

Tel. (55) 5729-6000, Ext. 73264 Fax (55) 56562058

e-mail: jlechugao1500@alumno.ipn.mx, ccruzra@ipn.mx, rreyesre@ipn.mx

Resumen- Con el auge del Internet de las cosas (IoT), la cantidad de dispositivos conectados a Internet, han aumentado de manera significativa, debido a su proliferación e inmersión en la vida cotidiana. Con lo cual se prevé que los sistemas que emplean esta tecnología prevalecerán en nuestra sociedad, lo cual dará lugar a sistemas de gran escala, generando nuevos desafíos en el manejo de la información, así como en su integridad. Esto conlleva a tomar conciencia sobre la seguridad y privacidad en las redes informáticas, donde estos dispositivos interactúan entre sí, debido a que estos convierten en un objetivo para ejecutar diversos ataques cibernéticos, como puede ser la suplantación de identidad, ataques de denegación de servicio (DDoS), entre otros. En el presente trabajo, se propone un sistema para la detección de anomalías en los datos transmitidos por dispositivos IoT, donde inicialmente se realiza un preprocesamiento de los datos, empleando la estimación de la densidad del núcleo (KDE), la estandarización y normalización (L1, L2) de los datos, posteriormente se utiliza un modelo de aprendizaje automático basado en la máquina de vectores de soporte (SVM) para la clasificación. Los resultados obtenidos demuestran una eficacia del sistema del 99.8%.

Palabras clave. Seguridad informática, Datos anómalos, IoT, Aprendizaje Automático, Máquina de Vectores de Soporte.

I. INTRODUCCIÓN

En los últimos años, los dispositivos IoT se han convertido en una de las principales tecnologías emergentes, con la cual se puede realizar una conexión a internet o con otros dispositivos físicos [1]. Según los informes proporcionados por Cisco, la cantidad de conexiones que usen la comunicación *Machine-to-Machine* (M2M), alcanzara la cantidad de 14,600 millones para el año 2022, de esta cantidad se espera que el 31% de estas conexiones sean efectuadas por dispositivos IoT [2]. Debido a los avances tecnológicos estos dispositivos cada vez son más usados en diferentes áreas, como los son, el sector industrial, la asistencia médica, los vehículos inteligentes, entre otros [3].

II. ANTECEDENTES

Jesus Lechuga, et al., [4] proponen un sistema, el cual, consiste en la recolección de diferentes datos, usando diferentes sensores (Magnético, ultrasónico, movimiento y un

fotorresistor), los cuales se encuentran conectados a varios dispositivos *NodeMCU*, estos dispositivos IoT, recolectan información durante un cierto periodo de tiempo, en una habitación para monitorear el comportamiento de una persona, posteriormente se realiza una inyección de datos anómalos de forma aleatoria, para simular el efecto de la intrusión de un agente externo. Para comprobar la veracidad de los datos, se hace uso de una Red Neuronal Artificial (RNA) Perceptrón Multicapa para identificar los datos anómalos enviados desde los dispositivos IoT.

Liang Liu, et al., [5] presentan un método para detectar nodos maliciosos en una red de sensores inalámbrica (WSN). En la cual, envían paquetes de información a los nodos de la red desde una fuente confiable. Cuando un paquete llega al receptor este lo analiza y extrae la ruta de transmisión junto con los valores que contiene para cada nodo, por donde se retransmite cada paquete. Seguido de esto, se hace uso de una regresión lineal con un gradiente descendiente (GD), para verificar la información, además de usar una máquina de vectores de soporte, para que esta aprenda a detectar los nodos confiables de acuerdo con la información de los paquetes recibidos.

Simon Duque, et al., [6] exponen un procedimiento para la detección de datos anómalos orientado al ámbito industrial, lo cual es llamado Internet Industrial de las Cosas (IIoT), enfocado a un sistema SCADA, haciendo uso del protocolo de comunicación *Modbus/TCP*, el cual es usado para transmisión de datos entre PLCs y unidades de terminales remotas (RTU). Al hacer uso de diferentes técnicas de Machine Learning (ML) para la detección de datos anómalos, dando resultados favorables en la detección de anomalías en estos dispositivos, ofreciendo una mejor seguridad ante ataques cibernéticos.

III. MÉTODO PROPUESTO

En la realización del presente trabajo, en primer lugar, se hace uso de la base de datos (BD) empleada en [4], con la intención de verificar la eficacia entre métodos de aprendizaje

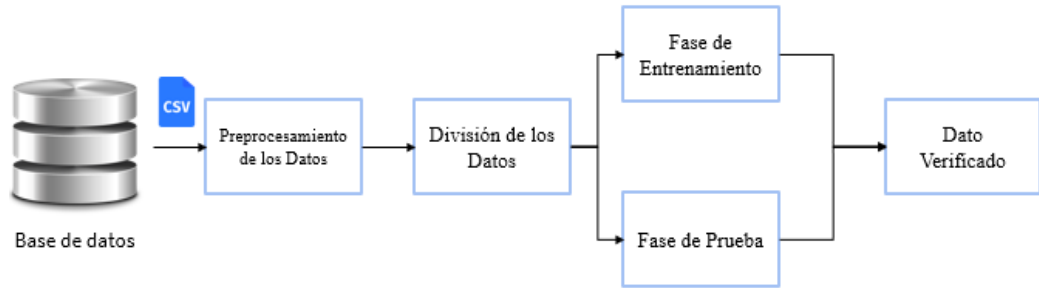


Fig. 1. Diagrama a bloques general del método propuesto

automático, en segundo lugar, se examina la parte de preprocesamiento de datos, en tercer lugar, se realiza el entrenamiento con diferentes arquitecturas propuestas, para la SVM, finalmente, se discute el modelo de aprendizaje automático empleado. En la Fig. 1 se muestra el diagrama a bloques general del método propuesto

A. Base de datos

La BD se encuentra constituida por 4000 muestras, recopiladas por los sensores empleados en [4], la cual consta de 6 campos, los cuales se mencionan a continuación. *ID*: identificador del dispositivo, *Hora*: tiempo en que se registró la información en la BD, *Puerta*: indica si el sensor magnético se encuentra activado o en reposo, *Distancia*: señala la distancia entre la posición del sensor ultrasónico y la pared (o si alguna persona entra en contacto con él), *Movimiento*: detecta movimientos en la habitación, *Luz*: indica si alguna fuente de luz emite energía, *Veracidad*: el cual señala si el conjunto de todos estos datos es veraz o no; para comprobar la veracidad del vector, se sigue una secuencia lógica, de las posibles acciones que realice una persona en su habitación. Los campos que contiene la BD se ilustran en la Tabla I. Posteriormente se exporta la BD a un archivo *comma-separated values* (CSV), el cual es un documento de formato abierto, el cual se emplea para representar los datos en forma de tabla, en donde cada columna se separa por comas [7].

TABLA I. CAMPOS ALMACENADOS EN LA BASE DE DATOS

ID	Hora	Sensores				
		Puerta	Distancia	Movimiento	Luz	Veracidad
313	20:30:45	0	100	0	0	1
2217	22:39:05	1	76	0	0	0
3867	00:32:15	1	0	0	1	0

B. Preprocesamiento de los datos

El preprocesamiento de datos descarta los campos *ID* y *Hora*, debido a que el primer campo, solo se usa como el número de la fila, mientras que el segundo no es contemplado en [4]. En la Tabla II, se muestran las características que se usarán para la clasificación de los datos atípicos, el campo *Veracidad*, indicara si los datos recibidos son verídicos o no. Para visualizar los datos se hace uso de la estimación de la

densidad del núcleo (KDE) propuesta por Murray Rosenblatt [8].

TABLA II. CARACTERÍSTICAS EMPLEADAS PARA LA CLASIFICACIÓN

Puerta	Distancia	Sensores		
		Movimiento	Luz	Veracidad
0	100	0	0	1
1	76	0	0	0
1	0	0	1	0

El cual es un método no paramétrico, que permite estimar la función de densidad de probabilidad (FDP), de una variable aleatoria, mediante un número finito de muestras como se muestra en (1), es usado debido a que uno de los inconvenientes de los histogramas, es que dependen de una gran cantidad de datos por intervalo.

$$f(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x-x_i}{h}\right) \quad (1)$$

donde x es la variable aleatoria, x_i es una muestra de un conjunto de datos, n es el número de observaciones, h es el ancho de banda y K es un Kernel.

Para K se usa un Kernel *Gaussiano* expresado en (2) y uno *Lineal* como en (3), para visualizar los datos como se muestra en la Fig. 2.

$$K(x; h) = \exp\left(-\frac{x^2}{2h^2}\right) \quad (2)$$

$$K(x; h) = 1 - \frac{x}{h}, \text{ si } x < h, \quad (3)$$

seguido de esto, se realiza un tratado de datos, para estandarizar z empleando (4), eliminando la media μ y escalando su varianza σ de forma que sea igual a 1 [9].

$$z = \frac{x-\mu}{\sigma} \quad (4)$$

Además de usar una normalización L1 y L2 representados por (5-6).

$$L1 = z = \|x\|_1 = \sum_{i=1}^n |x_i| \quad (5)$$

$$L2 = z = \|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2} \quad (6)$$

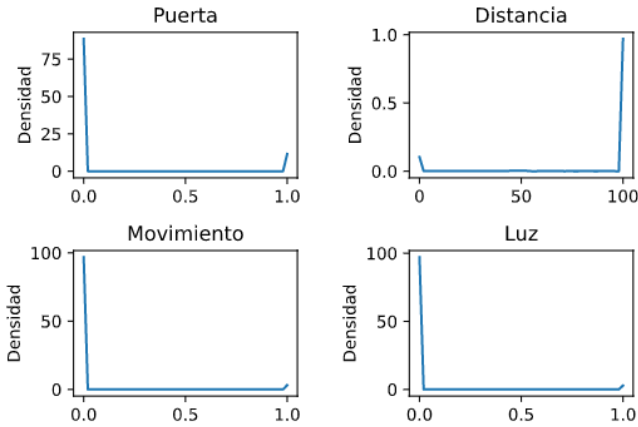


Fig. 2. FDP de los parámetros en la BD.

C. División de datos

La división de los datos se efectúa de forma aleatoria en dos conjuntos, entrenamiento y prueba, los cuales se dividen en una proporción de 80% y 20%, donde el 80% de los datos son para el entrenamiento del modelo, mientras que el 20% restante se reserva para la fase de prueba.

D. Fase de entrenamiento

En la fase entrenamiento, se usan las características mostradas en la Tabla II, donde las primeras 4 (puerta, distancia, movimiento y luz), son los datos de entrenamiento del clasificador, mientras que la última característica (veracidad), se emplea como etiqueta para verificar si los datos son verídicos. Como clasificador se utiliza una máquina de vectores de soporte (SVM), el cual es un método de aprendizaje supervisado. La SVM fue propuesta por primera vez por Vapnik [10] y construye un hiperplano o un conjunto de hiperplanos en un espacio dimensional alto o infinito. El objetivo principal de este modelo es identificar el hiperplano de margen máximo, el cual se define por la ubicación de los vectores de soporte, donde se maximice la separación entre clases, como se muestra en la Fig. 3. Existen dos categorías principales para las SVM, las cuales son: *support vector classification* (SVC), además de *support vector regression* (SVR).

Con el fin de encontrar el margen máximo de un hiperplano. Se asume que $x_i \in \mathbb{R}^d, i = 1, 2, \dots, N$, donde x_i forma un conjunto de vectores con sus correspondientes etiquetas $y_i, i = 1, 2, \dots, N$. La SVM puede mapear los vectores de entrada x_i en un espacio de características de altas dimensiones. Haciendo

uso de un Kernel $K(x_i, x_j)$, el cual realiza el mapeo, utilizando (7).

$$f(x) = \text{sgn}(\sum_{i=1}^N y_i \alpha_i * K(x, x_i) + b) \quad (7)$$

donde b es un término independiente.

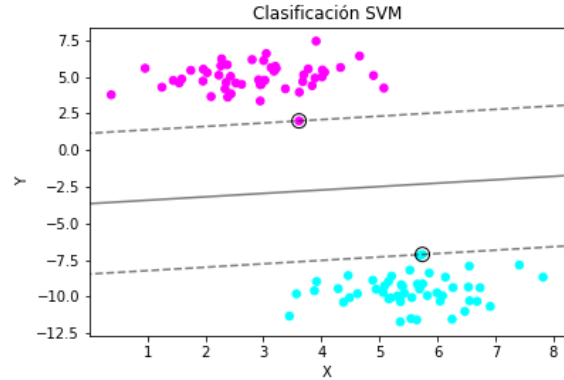


Fig. 3. Clasificación de dos clases usando SVM.

Es aquí donde se presenta el problema de programación cuadrática mostrado en (8-10) para valores de α_i , el objetivo de α es elegir la mejor aproximación para el clasificador, entre los datos de entrada x_i y su respectiva respuesta y_i .

$$\text{Max } \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j * y_i y_j * K(x_i, x_j), \quad (8)$$

$$\text{Sujeto a, } 0 \leq \alpha_j \leq c \quad (9)$$

$$\sum_{i=1}^N \alpha_i y_i = 0, \quad i = 1, 2, \dots, N \quad (10)$$

donde c es un parámetro de regularización.

Debido a que, en el presente trabajo, se busca hacer una clasificación (binaria) entre 2 clases, se opta por el uso de una SVC [11], puesto que se busca clasificar si los datos son verídicos o no. Una SVC recibe como parámetro de entrada los vectores $x_i \in \mathbb{R}^d, i = 1, 2, \dots, N$, en 2 clases, sus etiquetas ' y ' pueden ser $\{0, 1\}$ o $\{-1, 1\}$, donde el objetivo es encontrar $\alpha \in \mathbb{R}^d$ junto con una $b \in \mathbb{R}^d$, tal que la predicción dada por (11) sea correcta para la mayor cantidad de muestras.

$$f(x) = \text{sgn}(\alpha^T \phi(x) + b) \quad (11)$$

donde $\phi(x)$ es un espacio de características de alta dimensión y b es un término independiente.

La SVC resuelve el principal problema de programación cuadrática mostrado en (12-14) para valores de α .

$$\text{min } \frac{1}{2} \sum_{i=1}^N \alpha^T \alpha + c \sum_{i=1}^N \zeta_i \quad (12)$$

$$\text{Sujeto a, } y_i(\alpha^T \phi(x_i) + b) \geq 1 - \zeta_i, \zeta_i \geq 0 \quad (13)$$

$$\sum_{i=1}^N y_i(\alpha^T \phi(x_i) + b) = 0, \quad i = 1, 2, \dots, N \quad (14)$$

donde ζ es la distancia entre muestras y c es un parámetro de regularización.

Una vez resuelto este problema de optimización solo queda obtener la función de decisión (15) para cada muestra.

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b \quad (15)$$

donde K es un Kernel.

Para el Kernel se usan las siguientes funciones (16-19).

$$\text{Lineal: } K(x_i, x) = (x_i, x) \quad (16)$$

$$\text{Polinomial: } K(x_i, x) = (\gamma(x_i, x) + r)^d \quad (17)$$

$$\text{Base Radial: } K(x_i, x) = \exp(-\gamma \|x_i - x\|^2 + r) \quad (18)$$

$$\text{Sigmoide: } K(x_i, x) = \tanh(\gamma(x_i, x) + r) \quad (19)$$

donde d es el grado para la función polinomial y γ es una constante para la función de base radial.

E. Fase de prueba

Una vez que se tiene entrenada la SVC, se pone a prueba con el conjunto de datos reservados para comprobar el rendimiento del sistema.

IV. EXPERIMENTACIÓN Y RESULTADOS

El entrenamiento de la SVC, se hace realiza con la BD usada en [4], primeramente, el entrenamiento se hace sin el preprocesamiento de los datos, posteriormente, se implementa el procesamiento mencionado en la sección III, creando 3 conjuntos de datos diferentes. Se usa una tolerancia e , la cual corresponde a un parámetro de parada. El cual ayuda al problema de optimización, si no encuentra una mejora en el clasificador, este se detendrá. Los parámetros propuestos para la SVC se muestran en la Tabla III.

TABLA III. PARÁMETROS USADOS PARA EL ENTRENAMIENTO DE LA SVC

Parámetros	Lineal	Polinomial	Base Radial	Sigmoide
d	-	1,2,4	-	-
γ	-	-	0.5,1,5	0.5,1,5
c	1,5,10	1,5,10	1,5,10	1,5,10
e	0.001	0.001	0.001	0.001

Para evaluar el rendimiento de los clasificadores, se hace uso de las mediciones *Precision*, *Recall*, *F1-score*, *Support* (20-22).

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (20)$$

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (21)$$

$$F1 - \text{Score} = (1 + \beta^2) \frac{\text{Precision} * \text{Recall}}{(\beta^2 * \text{Precision}) + \text{Recall}} \quad (22)$$

donde TP es el Número de Verdaderos positivos, FP es el Número de Falsos positivos, FN es el Número de Falsos Negativos y β es el factor de importancia de *Precision*.

Precision es la capacidad del clasificador de no etiquetar como positiva una muestra negativa, *Recall* es la capacidad del clasificador de encontrar todas las muestras positivas, *F1-score* se puede interpretar como una media armónica de *Precision* y *Recall*, donde una puntuación *F1-score* igual a 1 se encuentra en su mejor valor, *Support* es el número de ocurrencias de cada clase [12], en las Tablas IV-VI, se muestran las arquitecturas con los mejores resultados obtenidos, para esto se realizan múltiples entrenamientos además de permutar los parámetros para la SVC propuestos en la Tabla III.

TABLA IV. SVC SIN TRATAMIENTO DE DATOS

Parámetros	Lineal $c = 1$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

Parámetros	Polinomial $c = 5, d = 4$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

Parámetros	Base Radial $c = 1, \gamma = 0.5$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

Parámetros	Sigmoide $c = 5, \gamma = 1$			
	Precision	Recall	F1-Score	Support
Anómalo 0	1.00	0.69	0.82	160
Veraz 1	0.93	1.00	0.96	640

TABLA V. SVC CON DATOS ESTANDARIZADOS

Parámetros	Lineal $c = 1$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

Parámetros	Polinomial $c = 1, d = 1$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

Parámetros	Base Radial $c = 1, \gamma = 0.5$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

Parámetros	Sigmoide $c = 5, \gamma = 0.5$			
	Precision	Recall	F1-Score	Support
Anómalo 0	0.99	1.00	1.0	160
Veraz 1	1.00	1.00	1.0	640

TABLA VI. SVC CON DATOS NORMALIZADOS (L2)

Parámetros	Lineal $c = 1$			
	Precision	Recall	F1-Score	Support
Anómalo 0	1.00	0.42	0.60	160
Veraz 1	0.87	1.00	0.93	640

Parámetros	Polinomial $c = 10, d = 4$			
	Precision	Recall	F1-Score	Support
Anómalo 0	1.00	0.69	0.82	160
Veraz 1	0.93	1.00	0.96	640

Parámetros	Base Radial $c = 5, \gamma = 5$			
	Precision	Recall	F1-Score	Support
Anómalo 0	1.00	0.69	0.82	160
Veraz 1	0.93	1.00	0.96	640

Parámetros	Sigmoide $c = 1, \gamma = 0.5$			
	Precision	Recall	F1-Score	Support
Anómalo 0	1.00	0.42	0.60	160
Veraz 1	0.87	1.00	0.93	640

En las Tablas VII-VIII, se muestra la Precisión obtenida, así como el Recall, para cada clasificador seleccionado, para discernir entre los valores atípicos, usando el conjunto de datos reservado para pruebas.

TABLA VII. PRECISIÓN OBTENIDA POR LOS CLASIFICADORES

Observaciones	Lineal	Polinomial	Base Radial	Sigmoide
Sin Tratamiento	0.9987	0.9987	0.9987	0.93875
Estandarizados	0.9988	0.9988	0.9988	0.9988
Normalizados	0.885	0.93875	0.93875	0.885

TABLA VIII. RECALL OBTENIDO POR LOS CLASIFICADORES

Observaciones	Lineal	Polinomial	Base Radial	Sigmoide
Sin Tratamiento	1.00	1.00	1.00	0.845
Estandarizados	1.00	1.00	1.00	1.00
Normalizados	0.71	0.845	0.845	0.71

En la Tabla IX, se muestra el mejor resultado obtenido entre las arquitecturas propuestas, frente a los resultados de [4].

TABLA IX. COMPARACION ENTRE ARQUITECTURAS

Arquitecturas	Precision	Recall
Lechuga [4]	0.995	1.00
Propuesto	0.9988	1.00

La Tabla X, muestra la matriz de confusión para la SVC con Kernel Lineal y datos estandarizados, todas las SVC con los parámetros mostrados en Tabla V, dan como resultado la misma matriz de confusión.

TABLA X. CLASIFICACIÓN DE LOS DATOS ANOMALOS POR LA SVC CON DATOS ESTANDARIZADOS

Matriz de Confusión		
Observaciones	Positivos	Negativos
Positivos	160	0
Negativos	1	639

V. CONCLUSIONES

Considerando los resultados obtenidos, se puede concluir, que el mejor desempeño del sistema propuesto se obtiene, cuando se realiza un preprocesamiento de los datos, en especial cuando los datos se estandarizan, como se muestra en la Tabla V, eliminando la media y escalando los datos de forma que su varianza sea igual a 1. Es de especial interés observar, que al efectuar un preprocesamiento de los datos además de emplear diferente Kernel, el clasificador SVC, obtiene la misma Precision para clasificar los datos anómalos de forma eficiente, esto se debe a que este tipo de arquitecturas funcionan mejor con valores de entrada que se encuentren entre [0, 1]. Con esta arquitectura se obtuvo un 0.003% más de Precision frente a [4], logrando el mismo objetivo, detectar los datos anómalos en dispositivos IoT. De igual manera, se puede mejorar de forma considerable este tipo de sistemas, escalando verticalmente estos sistemas, añadiendo diferentes sensores para obtener más información del entorno, además del uso de validaciones cruzadas, garantizando que la partición entre los datos de entrenamiento y de prueba son independientes entre sí.

AGRADECIMIENTOS

Agradecemos al Consejo Nacional de Ciencia y Tecnología, al Instituto Politécnico Nacional, a la COFAA del IPN y al Programa Institucional de Formación de Investigadores del IPN por el apoyo otorgado para el desarrollo de este trabajo.

REFERENCIAS

- [1] L. Atzori, A. Iera and G. Morabito, "The Internet of Things: a survey," Computer Networks, vol. 54, no. 15, pp. 2787-2805, 2010.
- [2] Cisco, "Cisco Visual Networking," Cisco public, 2017.
- [3] I. U. Din, M. Guizani, J. J.P.C. Rodrigues, S. Hassan and V. V. Korotaev, "Machine learning in the Internet of Things: Designed techniques for smart cities," Future Generation Computer Systems, vol. 100, pp. 826-843, 2019.
- [4] J. Lechuga, F. Sánchez, C. Cruz and R. Reyes, "Red Neuronal para la Detección de Datos Anómalos en Dispositivos IoT," Congreso Internacional de Robótica y Computación, vol. VII, pp. 153-157, 2020.
- [5] L. Liu, J. Yang and W. Meng, "Detecting malicious nodes via gradient descent and support vector machine in Internet of Things," Computers & Electrical Engineering, vol. 77, pp. 339-353, 2019.
- [6] S. D. Anton, S. Kanoor, D. Fraunholz and H. D. Schotten, "Evaluation of Machine Learning-based Anomaly Detection Algorithms on an Industrial Modbus/TCP Data Set," Proceedings of the 13th International Conference on Availability, Reliability and Security, p. 1-9, 2018.

- [7] Y. Shafranovich, "IETF Tools," SolidMatrix Technologies, Inc, October. 2005. Available <https://tools.ietf.org/html/rfc4180>.
- [8] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *Annals of Mathematical Statistics*, pp. 832-837, 1956.
- [9] P. L. Meyer, *Introductory probability and statistical application*, Addison-Wesley Publishing Company, 1970.
- [10] V. Vapnik, "An overview of statistical learning theory," *IEEE transactions on neural networks*, vol. 10, pp. 988-999, 1999.
- [11] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273-297, 1995.
- [12] D. Cournapeau, "<https://scikit-learn.org/>," Scikit-learn, 2020. Available: https://scikitlearn.org/stable/modules/generated/sklearn.metrics.classification_report.html?highlight=classification_report.
- [13] J. Lechuga, F. Sánchez, C. Cruz and R. Reyes, "<https://github.com>," GitHub, 2020. Available: <https://github.com/aguheel/Base-de-datos-Anomalos>.

Realización en FPGA de Sistema Encriptación de Imágenes con Atractores Caóticos: Dos-Alas y Jerk

Jose Cruz Nuñez Perez
Miguel Angel Estudillo Valdez,
and Andres Calvillo Tellez
Instituto Politécnico Nacional
IPN-CITEDI
Tijuana, Baja California, México, C. P. 22435
Email: jnunez@ipn.mx, calvillo}@citedi.mx

Alejandro Muñoz Organes
and Gilberto Enrico Vazquez Alcaraz
Tecnológico Nacional de México
Instituto Tecnológico de Tijuana
Tijuana, Baja California, México, C. P. 22430
Email: munoz.alejandro101@gmail.com,
gilberto.vazquez@tectijuana.edu.mx

Resumen—En este artículo se lleva a cabo la implementación en FPGA de un sistema de encriptación de información utilizando caos. Primeramente se hace uso de los métodos numéricos discretos de Euler Mejorado y Runge-Kutta de cuarto orden para resolver las ecuaciones diferenciales que describen el modelo de los osciladores caóticos. Posteriormente se compara el desempeño y utilización de recursos lógicos a fin de encontrar el método numérico que sea más eficiente debido a la cantidad limitada de recursos que tiene un FPGA. Las aportaciones de este trabajo son un simulador y un sistema de encriptación de imágenes modular con coeficientes de correlación de hasta 4% y un bajo consumo de recursos lógicos.

Palabras clave—Caos, Dos-Alas, Encriptación, Euler, FPGA, Jerk, Runge-Kutta

I. INTRODUCCIÓN

La teoría del caos es una rama de las matemáticas que se centra en el estudio de sistemas dinámicos cuyos estados aparentemente aleatorios se rigen por leyes deterministas que son muy sensibles a las condiciones iniciales. Pequeñas diferencias en las condiciones iniciales, como las causadas por errores de redondeo en el cálculo numérico, pueden producir resultados muy diferentes, por lo que predecir su comportamiento a largo plazo es prácticamente imposible.

En 1960 el meteorólogo Edward N. Lorenz del MIT, siendo escéptico de los modelos lineales utilizados para predecir el clima, decidió desarrollar un modelo que describe el comportamiento del clima utilizando doce ecuaciones que involucraban factores meteorológicos relevantes como la temperatura, presión del aire y la velocidad del viento [1]. Durante sus experimentos, Lorenz descubrió que al reingresar en su computadora los parámetros para ejecutar la simulación, a partir de la mitad, la predicción obtenida era completamente diferente a la obtenida anteriormente cuando se ejecuta desde el principio. El error de redondeo de decimales fue el culpable de este cambio de comportamiento, con ello Lorenz había descubierto un sistema que era altamente sensible a las condiciones iniciales. Esta propiedad es matemáticamente descrita mediante el Exponente de Lyapunov, en donde una ecuación tiene tantos exponentes de Lyapunov como variables de estado, y basta con que uno de ellos sea positivo para que

el sistema sea considerado como caótico [2].

La mayoría de los sistemas de encriptación utilizados actualmente están basados en llaves, secuencias de caracteres que controlan un proceso o algoritmo de encriptación. Debido a su naturaleza es necesario recurrir a una práctica de cambio constante de llaves para mitigar ataques de “fuerza bruta”, que pueden deducirlas en tiempo polinomial si la llave es constante y se tiene una cantidad indefinida de intentos. Por lo cual se requieren nuevas formas de encriptación que sean resistentes a los ataques de fuerza bruta y que mantengan un nivel alto de complejidad.

El desarrollo y la implementación de sistemas en arreglo de compuertas lógicas programables en campo (FPGA) ofrecen la oportunidad de probar un diseño antes de su implementación en Circuitos Integrados de Aplicación Específica (ASIC, por sus siglas en inglés, *Application Specific Integrated Circuit*). No obstante, como en cualquier sistema digital se tiene el problema inherente del error causado por la conversión de sistemas continuos a discretos y viceversa que se reduce con sistemas de conversión más complejos. Sin embargo, un FPGA tiene una cantidad limitada de recursos que se deben tomar en consideración durante el diseño.

En los últimos años se han publicado diversas investigaciones enfocadas al caos, sus implementaciones digitales y posibles aplicaciones especialmente en el área de telecomunicaciones con el objetivo de conseguir sistemas de encriptación de información. En [3] se realizaron experimentos de comunicación segura utilizando osciladores caóticos sincronizados para la transmisión y recepción, utilizando el enfoque de generación de secuencias pseudo-aleatorias para enmascarar información. Actualmente se ha logrado encriptar y desencriptar información en sistemas de comunicación serial usando osciladores caóticos en FPGA, utilizando estrategias de generación de llaves por medio de mapas caóticos y de encriptación de información por adición de caos. Por lo que se comprueba en esta última

que la información encriptada generada posee coeficientes de correlación incluso menores al 1% con respecto a la información original [3] [4]. Además, se han encontrado estrategias que reducen la degradación dinámica causada por la precisión finita y cuantización de los sistemas caóticos para su conversión digital [5].

Para propósitos de este documento se consideran dos osciladores de tipo interruptor: 1) el oscilador caótico de Dos-Alas (*Two – Wing*) y Jerk de Interruptor-Único (*Single – Switch Jerk*); en el segundo caso se utiliza la función interruptor una sola vez, a diferencia del Dos-Alas que utiliza la función en dos ocasiones. En el artículo [6] se realizaron experimentos de análisis del rendimiento a los dos tipos de sistemas caóticos tipo interruptor en FPGA: Jerk de interruptor único y de Dos-Alas.

Este artículo está organizado de la siguiente manera: en la sección II se describen los sistemas caóticos que se utilizaron para esta investigación, se introducen los conceptos de sincronización y encriptación de información, además de hacer un breve repaso sobre osciladores y bloques en VHDL. En la sección III se muestran los resultados de las simulaciones realizadas en Matlab, así como la descripción del comportamiento del error obtenido para cada ciclo de iteraciones en los métodos numéricos de Euler y Runge Kutta de cuarto orden. La sección IV presenta los resultados obtenidos del sistema para encriptar imágenes utilizando tarjetas de FPGA. Finalmente, las conclusiones se presentan en la sección V.

II. DESARROLLO

A. Osciladores Tipo Interruptor

El primer oscilador caótico es el Dos-Alas [6], que se describe por el sistema de ecuaciones diferenciales (1),

$$\begin{aligned} \frac{dx}{dt} &= a(y - x) \\ \frac{dy}{dt} &= (b - z)f(x), \\ \frac{dz}{dt} &= xf(x) - cz \end{aligned} \quad (1)$$

donde a, b y c son coeficientes constantes en el intervalo [0 1], mientras que $f(x)$ corresponde a la ecuación (2),

$$f(x) = \begin{cases} S_1, & x < B \\ S_2, & x \geq B \end{cases} \quad (2)$$

donde S_1 y S_2 representan los niveles de cambio y B el punto de quiebre, es decir, la función alternará entre S_1 y S_2 dependiendo del valor de B .

El segundo oscilador es el Jerk de Interruptor-Único, a diferencia del Dos-Alas este oscilador utiliza la función interruptor

una sola vez y su comportamiento viene descrito mediante el sistema de ecuaciones diferenciales (3),

$$\begin{aligned} \frac{dx}{dt} &= y \\ \frac{dy}{dt} &= z \\ \frac{dz}{dt} &= -[ax + bz + cyf(y)] \end{aligned} \quad (3)$$

donde a, b y c son coeficientes constantes con valor de 1, mientras que $f(y)$ corresponde a la ecuación (4),

$$f(y) = \begin{cases} S_1, & x < B \\ S_2, & x \geq B \end{cases}, \quad (4)$$

donde S_1 y S_2 representan los niveles de cambio y B el punto de quiebre.

B. Sincronización Hamiltoniana

Se considera que dos sistemas están sincronizados cuando el comportamiento de estos tiene una evolución idéntica al paso del tiempo. El hecho de poder sincronizar dos sistemas distintos abre la posibilidad de estudiar un sistema desconocido a partir de uno conocido [7] [8]. Un sistema caótico es altamente sensible a las condiciones iniciales, esto implica que dos trayectorias que surgen de condiciones iniciales cercanas pero diferentes se separan exponencialmente al paso del tiempo. Como resultado, los sistemas caóticos desafían la sincronización, porque dos sistemas idénticos que parten de condiciones iniciales ligeramente diferentes evolucionarían en el tiempo de una forma no sincronizada. Por ello es necesario una estrategia que mantenga la sincronización y que sea independiente de las condiciones iniciales, ello implica un canal de “comunicación” o bien, un adecuado acoplamiento entre ambos sistemas para una retroalimentación activa.

Por medio del método de Hamilton es posible llevar a cabo la sincronización de dos sistemas caóticos en configuración maestro-esclavo. Para ello, se toma un sistema dinámico de la forma [9] [10]:

$$\dot{x} = f(x), \quad (5)$$

que puede ser reescrito como

$$\dot{x} = A \frac{\partial H}{\partial x} + F(x), \quad (6)$$

y debido a la siguiente propiedad de matrices:

$$A = \frac{A - A^T}{2} + \frac{A + A^T}{2}, \quad (7)$$

es posible reescribir (6) al sustituir A por (7) y expandirla a (8)

$$\dot{x} = \frac{A - A^T}{2} \frac{\partial H}{\partial x} + \frac{A + A^T}{2} \frac{\partial H}{\partial x} + F(x). \quad (8)$$

Con el fin de simplificar se considera S una matriz simétrica constante e I una matriz antisimétrica constante,

$$J(x) = \frac{A - A^T}{2}, \quad (9)$$

$$S(x) = \frac{A + A^T}{2}, \quad (10)$$

Se sustituyen (9) y (10) en (8) para ser reescrita en la forma Hamiltoniana generalizada canónica,

$$\dot{x} = J(x) \frac{\partial H}{\partial x} + S(x) \frac{\partial H}{\partial x} + F(x), \quad (11)$$

donde $H(x)$ es una función de energía suave globalmente positiva en R^n ,

$$H(x) = \frac{1}{2} x^T M x, \quad (12)$$

donde M es una matriz constante, simétrica y positiva. En tal caso, $\partial H / \partial x = Mx$. Las matrices cuadradas $J(x)$ y $S(x)$ satisfacen para toda $x \in R^n$ las siguientes propiedades:

$$J(x) + J^T(x) = 0, \quad (13)$$

$$S(x) - S^T(x) = 0, \quad (14)$$

La salida del sistema está dada por la ecuación:

$$y = C \frac{dH}{dx}, \quad (15)$$

donde C es una matriz constante. El valor estimado de $x(t)$ es $\varepsilon(t)$, por lo que la función de energía hamiltoniana se representa como $H(\varepsilon)$.

$$\varepsilon = J(y) \frac{dH}{d\varepsilon} + (I + S) \frac{dH}{d\varepsilon} + f(y) + K(y - n), \quad (16)$$

donde $n = C \frac{dH}{d\varepsilon}$ y K es la ganancia constante del sistema.

El error de estimación se define como $e(t) = x(t) - \varepsilon(t)$. La estimación del error se calcula utilizando la forma hamiltoniana generalizada

$$e = J(y) \frac{dH}{de} + (I + S - KC) \frac{dH}{de}. \quad (17)$$

Para ambos tipos de osciladores, la implementación se divide en dos secciones, la primera es el oscilador maestro y la segunda el oscilador esclavo. El oscilador maestro es un oscilador “común”. El caso del oscilador esclavo es diferente, debido a que la sincronización requiere de retroalimentar el esclavo por cada iteración.

C. Encriptación

Como método de encriptación se utiliza la compuerta X-OR, este método consiste en hacer la operación X-OR bit por bit, de la información con una llave. Cuando se aplica el mismo procedimiento al resultado, usando la misma llave, se obtiene de nuevo la información original, por lo que no es necesario implementar dos procesos distintos, uno para encriptar y otro para desencriptar. Conociendo lo anterior, se implementa una función en MATLAB llamada **crypto_xor** que consiste en:

- Separar la imagen RGB en tres matrices, una para cada canal.

- Convertir el canal a binario, este proceso convierte cada dato (pixel) de la matriz a un byte, con cada bit separado, resultando en una matriz de [cantidad de pixeles] \times 8.
- Se convierte el binario del canal a tipo lógico, esto debido a que no es posible utilizar operadores lógicos con el tipo binario por sí solo.
- Se convierte el vector que contiene la señal caótica a binario, resultando en una matriz igual al canal.
- Se convierte el vector caótico a lógico.
- Se realiza la operación X-OR bit por bit entre los elementos del vector caótico y del canal de la imagen, se guarda resultado en nueva matriz.
- La matriz resultante se convierte de lógico a decimal.
- Las dimensiones de la matriz resultante se reajustan para ser iguales que la imagen original.
- Repetir los pasos 2 a 8 para cada canal RGB.
- Concatenar las tres matrices resultantes en una sola matriz de tres dimensiones.

La función **crypto_xor** recibe de entrada cuatro parámetros:

- Una imagen RGB de dimensiones cuadradas.
- Tres vectores de una dimensión (x_0, y_0, z_0) que corresponden a la “llave” criptográfica, estos vectores deben tener un tamaño mínimo igual a la cantidad de pixeles que tiene la imagen.

Como salida se obtiene una imagen RGB con dimensiones iguales a la imagen de entrada y se muestran en la consola los coeficientes de correlación entre los canales de la imagen de entrada y de salida.

D. Osciladores en VHDL

En [6] se encontró que para lograr caos en el oscilador caótico de Dos-Alas se requieren 14 bits para la parte fraccionaria, 1 bit para el entero y 1 bit para el signo; en el caso del oscilador Jerk de interruptor único se requieren 12 bits para la parte fraccionaria, 3 bits para el entero y 1 bit para el signo. En este artículo se utilizará una notación de punto fijo de 18 bits con el fin de utilizar el mismo algoritmo para ambos osciladores, usando 14 bits para la parte fraccionaria, 3 bits para el entero y 1 bit para el signo. La implementación se llevó a cabo con una tarjeta de séptima generación de Xilinx. Las librerías que permiten la funcionalidad de punto flotante fueron **fixed_pkg** y **fixed_float_types**. Idealmente, los tipos de señal en la entidad deberían ser de uno de los tipos básicos del estándar (bit, STD_LOGIC, etc.) por motivos de compatibilidad con otras entidades. En virtud de que este será un sistema cerrado, todas las comunicaciones entre entidades se llevaron a cabo usando el tipo **sfixed**.

1) *Módulo de métodos numéricos:* La implementación en VHDL de los métodos numéricos RK4 y Euler Mejorado calcula el paso siguiente en base al paso anterior, pero no tiene memoria de cualquier paso anterior a $n - 1$. Con el enfoque elegido en este trabajo se realizan todas las aproximaciones

en un solo ciclo de reloj, esto tiene un impacto positivo en la velocidad de cálculo, pero negativo en el uso de recursos lógicos. Con el fin de facilitar el intercambio de paquetes y métodos numéricos se mantienen las mismas convenciones de nombres para las señales de entrada y salida de las entidades. La figura 1 muestra la entidad implementada en VHDL para los métodos numéricos RK4 y Euler Mejorado.

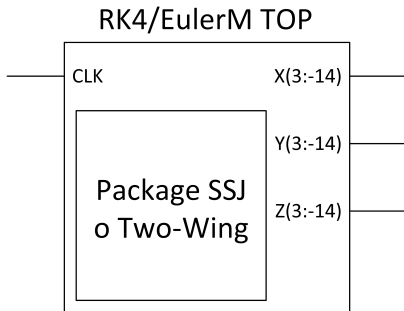


Fig. 1. Entidad de VHDL para los métodos numéricos RK4 y Euler Mejorado

2) *Módulo de sincronización*: El oscilador esclavo recibe en la retroalimentación con el error ya compensado, este valor se calcula por medio de un módulo intermedio, un “sincronizador”, este módulo utiliza la estrategia de sincronización Hamiltoniana. La figura 2 ilustra la entidad implementada en VHDL para el modulo de sincronización entre el maestro y el esclavo, es decir, el transmisor y receptor.

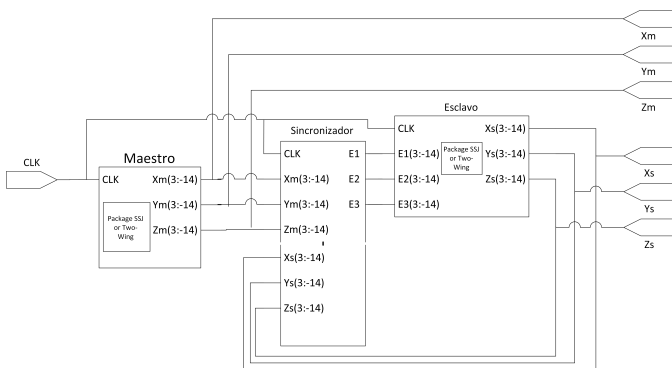


Fig. 2. Entidad de VHDL para el módulo de sincronización entre el maestro y el esclavo

3) *Módulo de Transmisión y Recepción de Datos*: En este módulo se unifican las implementaciones vistas anteriormente y se agrega el proceso de encriptación/desencryptación por medio de la comunicación UART. El algoritmo consiste en un proceso que se adjunta al módulo **top** para coordinar la comunicación UART, este consiste de tres estados: **wait_input**, **Encrypt** y **SEND**.

En el primer estado **wait_input**, el proceso se encuentra en un bucle infinito esperando un paquete de entrada desde el módulo UART, una vez que se recibe uno, el dato se guarda en memoria local y se indica al proceso que en el siguiente

ciclo debe saltar al estado **Encrypt**.

El estado **Encrypt** avisa al módulo UART que el mensaje fue descargado con éxito y procede a realizar la operación X-OR a cada uno de los 8 bits del mensaje con otros 8 bits provenientes de una de las señales caóticas del oscilador maestro y guarda el resultado en la variable **Encrypted**. Justo después, se realiza la operación X-OR bit por bit de la variable **Encrypted** con una de las señales caóticas del oscilador esclavo, obteniéndose de nuevo el mensaje original, el resultado de esta última operación se guarda en la variable **UART_OUT** y se indica al proceso que el siguiente ciclo de reloj proceda al estado **SEND**.

En el estado **SEND** se envía el mensaje **UART_OUT** a la entrada el módulo UART y se mantiene activo hasta que el módulo indique que ya lo ha descargado por medio de una bandera, si no se recibe esta bandera, el estado **SEND** se mantendrá en bucle con el mensaje activo en la entrada del módulo UART. Cuando el módulo indique que ha descargado el mensaje se indica al proceso que salte al estado **wait_input**, reiniciando la máquina de estados.

El algoritmo anterior se implementa en un módulo VHDL independiente, la Figura 3 muestra las conexiones de este módulo, denominado **TX/RX**, con el resto del sistema.

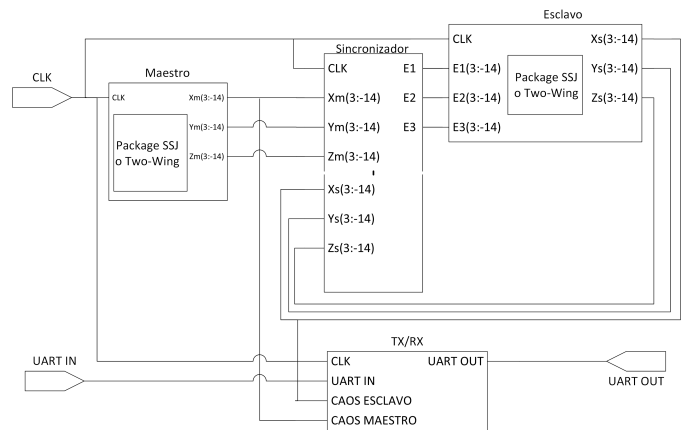
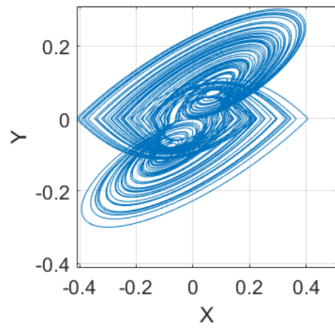


Fig. 3. Entidad de VHDL para la transmisión y recepción de datos

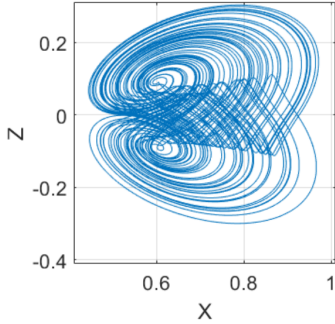
III. SIMULACIÓN

Al realizar la simulación en Matlab del sistema de ecuaciones del oscilador de Dos-Alas (1) con los parametros $a = b = 0.6$, $c = 0.15$, $S_1 = -1$, $S_2 = 1$, $B = 0$, y las condiciones iniciales $x(0) = 0.1$, $y(0) = z(0) = 0$ se obtienen las gráficas de la figura 4 en los planos a) $x - y$, b) $x - z$, c) $y - z$.

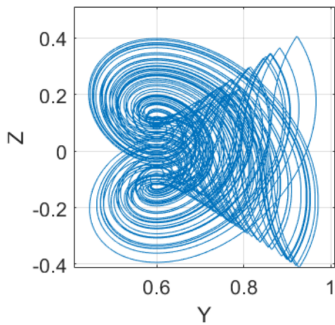
Para simular las trayectorias del sistema caótico Jerk de Interruptor-Único se resolvió el sistema de ecuaciones (3) en el software Matlab utilizando los parámetros $a = b = c = 1$, $S_1 = 5$, $S_2 = 0$, $B = 1$, y las condiciones iniciales $x(0) = 0.1$, $y(0) = z(0) = 0$, obteniendo así las gráficas de la figura 5 en los planos a) $x - y$, b) $x - z$, c) $y - z$.



(a)



(b)



(c)

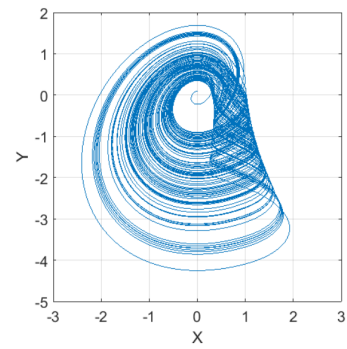
Fig. 4. Oscilador Caótico de Dos-Alas, a) Plano $x - y$, b) plano $x - z$, c) plano $y - z$ del oscilador.

A. Dos-Alas usando Euler Mejorado

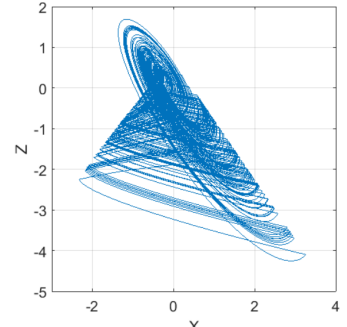
En el sistema maestro se utiliza el oscilador Dos-Alas con condiciones iniciales y coeficientes del párrafo anterior. El sistema esclavo utiliza el mismo oscilador caótico como observador, se mantienen los mismos coeficientes a , b y c , y los mismos parámetros para la función interruptor S_1 , S_2 y B , para corroborar sincronización se cambian las condiciones iniciales a $x(0) = 1.2$, $y(0) = 1.7$ y $z(0) = 1.7$. Además se eligen las ganancias $K_1 = 2$, $K_2 = 7$ y $K = 5$ para el observador.

B. Dos-Alas usando RK4

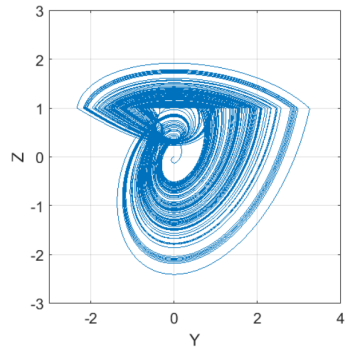
El sistema esclavo utiliza el mismo oscilador caótico como observador, se mantienen los mismos coeficientes a , b y c , y los mismos parámetros para la función interruptor S_1 , S_2 y



(a)



(b)



(c)

Fig. 5. Oscilador Caótico tipo *single-switch jerk*, a) Plano $x - y$, b) plano $x - z$, c) plano $y - z$ del oscilador.

B , para corroborar sincronización se cambian las condiciones iniciales a $x(0) = 1.2$, $y(0) = 1.7$ y $z(0) = 1.7$. Se eligen las ganancias $K_1 = 2$, $K_2 = 7$ y $K_3 = 5$ para el observador. El nuevo sistema de ecuaciones para una sincronización del sistema de Dos-Alas tanto para Runge Kutta cuarto orden como para Euler Mejorado es (18),

$$\begin{aligned} \frac{dx}{dt} &= a(y - x) + K_1(x_m - x) \\ \frac{dy}{dt} &= (b - z)f(x) + K_2(y_m - y), \\ \frac{dz}{dt} &= xf(x) - cz + K_3(z_m - z) \end{aligned} \quad (18)$$

En la figura 6 se muestran las trayectorias del sistema maestro y esclavo utilizando el método numérico Euler Mejorado. Mientras que en la figura 7 se muestran las trayectorias del sistema maestro y esclavo utilizando el método numérico Runge Kutta cuarto orden.

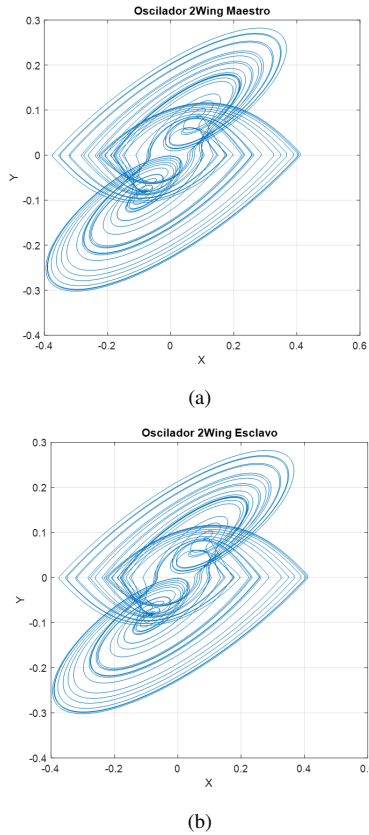


Fig. 6. Oscilador de Dos-Alas a) Maestro, b) Esclavo usando Euler Mejorado

C. Jerk de Interruptor Único usando Euler Mejorado

El sistema maestro utiliza el oscilador Jerk con condiciones iniciales y coeficientes descritos en los párrafos anteriores. El sistema esclavo utiliza el mismo oscilador caótico como observador, se mantienen los mismos coeficientes a , b , c y los mismos parámetros para la función interruptor S_1 , S_2 y B , para corroborar sincronización se cambian las condiciones iniciales a $x(0) = 1.2$, $y(0) = 1.7$ y $z(0) = 1.7$. Se eligen las ganancias $K_1 = 2$, $K_2 = 7$ y $K_3 = 5$ para el observador.

D. Jerk de Interruptor Único usando RK4

El sistema esclavo utiliza el mismo oscilador caótico como observador, se mantienen los mismos coeficientes a , b y c y los mismos parámetros para la función interruptor S_1 , S_2 y B , para corroborar sincronización se cambian las condiciones iniciales a $x(0) = 1.2$, $y(0) = 1.7$ y $z(0) = 1.7$. Se eligen las ganancias $K_1 = 2$, $K_2 = 7$ y $K_3 = 5$ para el observador. El nuevo sistema de ecuaciones para una sincronización *Single Switch Jerkal* tanto para Runge Kutta cuarto orden

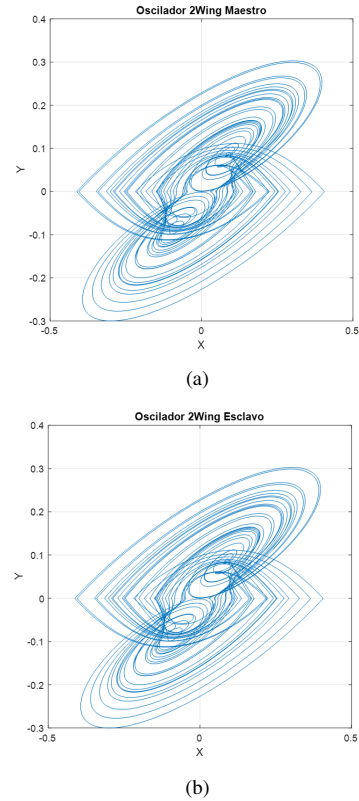


Fig. 7. Oscilador de Dos-Alas a) Maestro, b) Esclavo usando Runge Kutta de cuarto orden

como para Euler Mejorado es (19),

$$\begin{aligned} \frac{dx}{dt} &= y + K_1(x_m - x) \\ \frac{dy}{dt} &= z + K_2(y_m - y) \\ \frac{dz}{dt} &= -(ax + bz + cyf(y)) + K_3(z_m - z) \end{aligned} \quad (19)$$

En la figura 8 se muestran las trayectorias del sistema Jerk de Interruptor Único maestro y esclavo utilizando el método numérico Euler Mejorado. En la figura 9 se ilustran las trayectorias del sistema Jerk de Interruptor Único maestro y esclavo utilizando el método numérico Runge Kutta de cuarto orden.

IV. RESULTADOS

En esta sección se describen los resultados obtenidos en la implementación en FPGA. Primeramente, se realizó la encriptación y desencriptación de imágenes utilizando los osciladores caóticos sincronizados, el Dos-Alas y el Jerk de Interruptor Único, cuyos estados son calculados usando los métodos numéricos Euler mejorado y Runge-Kutta de cuarto orden. Para poder realizar las comparaciones entre los resultados obtenidos, se utilizó una imagen de 128×128 píxeles para todos los casos, esta imagen se muestra en la figura 10.

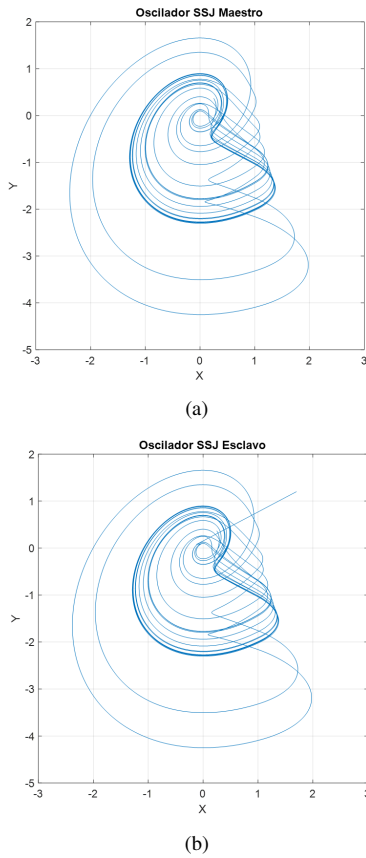


Fig. 8. Oscilador Jerk de Interruptor Único a) Maestro, b) Esclavo usando Euler Mejorado

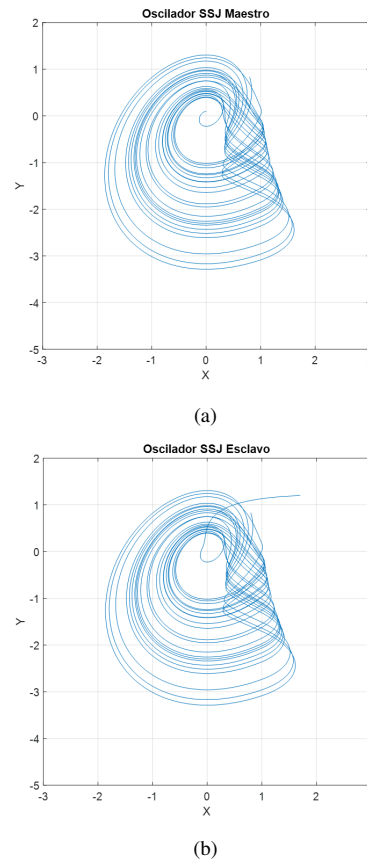


Fig. 9. Oscilador Jerk de Interruptor Único a) Maestro, b) Esclavo usando Runge Kutta de cuarto orden

A. Oscilador caótico Jerk de Interruptor Único

1) *Euler Mejorado*: Se configura el módulo **Top** para utilizar los módulos **EulerM** y **Slave_EulerM** que implementan el método numérico de Euler Mejorado tanto para el oscilador maestro y oscilador esclavo. A su vez se configuran los módulos **EulerM** y **Slave_EulerM**, para que utilicen el paquete del oscilador Jerk de Interruptor Único y se carga el proyecto el proyecto a la tarjeta de FPGA. Una vez que termine la transmisión, se obtiene la imagen encriptada de la Figura 11.

2) *Runge-Kutta de cuarto orden*: Se configura el módulo **Top** para utilizar los módulos **RK4** y **Slave_RK4** que implementan el método numérico de Runge-Kutta de cuarto orden tanto para el oscilador maestro y oscilador esclavo. A su vez se configuran los módulos **RK4** y **Slave_RK4**, para que utilicen el paquete del oscilador Jerk de Interruptor Único y se carga el proyecto el proyecto a la tarjeta de FPGA. Una vez que termine la transmisión, se obtiene la imagen encriptada de la Figura 12.

B. Oscilador caótico two-wing

1) *Euler Mejorado*: Se configura el módulo **Top** para utilizar los módulos **EulerM** y **Slave_EulerM** que implementan el método numérico de Euler Mejorado tanto para el oscilador



Fig. 10. Imagen RGB de 128×128 a encriptar con implementaciones en FPGA.

maestro y oscilador esclavo. A su vez se configuran los módulos **EulerM** y **Slave_EulerM** en si para que utilicen el paquete de Dos-Alas y se carga el proyecto el proyecto a la FPGA. Una vez que termine la transmisión, se obtiene la imagen encriptada de la Figura 13.

2) *Runge Kutta de cuarto orden*: Se configura el módulo **Top** para utilizar los módulos **RK4** y **Slave_RK4** que implementan el método numérico de Runge-Kutta de cuarto

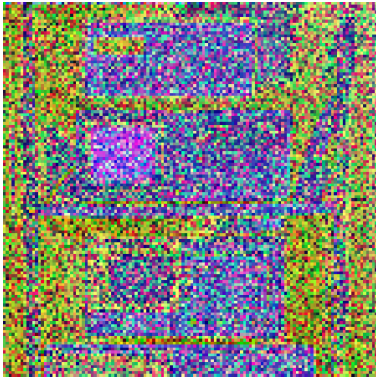


Fig. 11. Imagen encriptada por FPGA usando oscilador Jerk de Interruptor Único con Euler Mejorado.

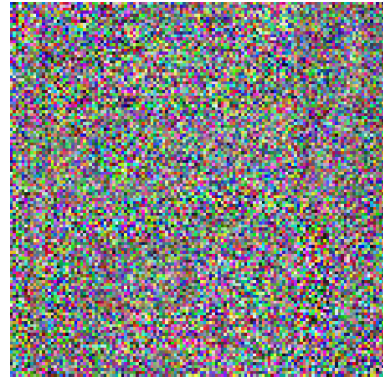


Fig. 13. Imagen encriptada por FPGA usando oscilador de Dos-Alas con Euler Mejorado.

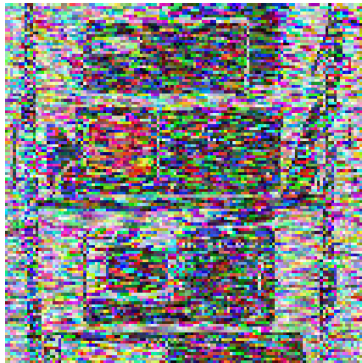


Fig. 12. Imagen encriptada por FPGA usando oscilador Jerk de Interruptor Único con Runge Kutta cuarto orden.



Fig. 14. Imagen encriptada por FPGA usando oscilador de Dos-Alas con Runge Kutta cuarto orden.

orden tanto para el oscilador maestro y oscilador esclavo. A su vez se configuran los módulos **RK4** y **Slave_RK4**, en si para que utilicen el paquete de *Two – Wing* y se carga el proyecto el proyecto a la FPGA. Una vez que termine la transmisión, se obtiene la imagen encriptada de la Figura 14.

Se calculó el coeficiente de correlación de cada canal de la imagen recibida con cada canal de la imagen original, usando los metodos de Euler y Runge Kutta de cuarto orden. La Tabla I muestra los resultados obtenidos para el oscilador Jerk de Interruptor Único, mientras que la Tabla II los del oscilador de Dos-Alas.

TABLA I
COEFICIENTES DE CORRELACIÓN DE IMAGEN ENCRYPTADA Y DESENCRIPTADA POR EL OSCILADOR JERK DE INTERRUPTOR ÚNICO USANDO LOS MÉTODOS DE EULER MEJORADO Y RUNGE KUTTA EN FPGA

Euler Mejorado			
Canal	Rojo	Verde	Azul
Coefficiente de correlación	0.2173	0.2266	0.9916
Runge-Kutta cuarto orden			
Coefficiente de correlación	0.42378	0.42826	0.39826

método numérico Euler Mejorado tiene el mejor desempeño entre los casos evaluados, con un coeficiente de correlación promedio (entre canales) de 0.054, seguido del Dos-Alas con RK4 con un coeficiente de correlación promedio de 0.0863. Ambos casos del oscilador Jerk de Interruptor Único mostraron los peores resultados con coeficientes de correlación promedio de 0.4785 con Euler Mejorado y 0.42 con RK4.

Este comportamiento se debe principalmente a la complejidad del oscilador de Dos-Alas con respecto al Jerk de Interruptor Único, ya que el este último solo tiene un elemento no lineal, mientras que Dos-Alas tiene dos, es decir, la cantidad de funciones interruptor que posee. El poseer dos funciones interruptor causa que las soluciones obtenidas en el oscilador

TABLA II
COEFICIENTES DE CORRELACIÓN DE IMAGEN ENCRYPTADA Y DESENCRIPTADA POR EL OSCILADOR DOS-ALAS USANDO LOS MÉTODOS DE EULER MEJORADO Y RUNGE KUTTA EN FPGA

Euler Mejorado			
Canal	Rojo	Verde	Azul
Coefficiente de correlación	0.0445	0.0591	0.0586
Runge-Kutta cuarto orden			
Coefficiente de correlación	0.094201	0.084856	0.080171

Se puede observar que el oscilador caótico de Dos-Alas con

TABLA III

USO DE RECURSOS DE FPGA DEL SISTEMA DE ENCRIPCIÓN USANDO OSCILADOR JERK DE INTERRUPTOR ÚNICO CON LOS MÉTODOS DE EULER MEJORADO Y RUNGE KUTTA.

Euler Mejorado			
Recurso	Utilización	Disponibles	Porcentaje de uso
LUT	849	133800	0.63%
Flip-Flop	129	267600	0.05%
DSP	12	740	1%
I/O	4	400	1%
Runge Kutta cuarto orden			
LUT	1973	133800	1.48%
Flip-Flop	129	267600	0.05%
DSP	19	740	2.57%
I/O	4	400	1%

TABLA IV

USO DE RECURSOS DE FPGA DEL SISTEMA DE ENCRIPCIÓN USANDO OSCILADOR DE DOS-ALAS CON LOS MÉTODOS DE EULER MEJORADO Y RUNGE KUTTA.

Euler Mejorado			
Recurso	Utilización	Disponibles	Porcentaje de uso
LUT	886	133800	0.66%
Flip-Flop	126	267600	0.05%
DSP	17	740	2.30%
I/O	4	400	1%
Runge Kutta cuarto orden			
LUT	1737	133800	1.30%
Flip-Flop	129	267600	0.05%
DSP	31	740	4.19%
I/O	4	400	1%

de Dos-Alas “orbiten” entre dos atractores, saltando de uno a otro de forma repentina a causa de la función interruptor, esto reduce la auto-correlación entre los valores adyacentes generados, y lo hace aún más sensible a los valores de las señales. Al ser el oscilador de Dos-Alas más sensible a los valores numéricos la introducción de error, causado por truncación de decimales o por el cálculo por medio de métodos numéricos, agrega un comportamiento no lineal adicional que lo vuelve mejora el comportamiento caótico. Es por ello que el método más inexacto, Euler Mejorado, muestra mejores resultados que el método RK4.

La Tabla III muestra la comparación del empleo de recursos lógicos de la implementación de FPGA del sistema de encriptación usando oscilador Jerk de Interruptor Único con los métodos de Euler Mejorado y Runge Kutta. Además la Tabla IV muestra los recursos lógicos utilizados por el oscilador de Dos-Alas.

Otro factor que establece al caso de Dos-Alas por Euler Mejorado como la mejor opción es el uso de recursos con una utilización promedio de 1% de la tarjeta AC701. Si bien, es su homólogo de Single Swich Jerk el que menos recursos lógicos utiliza con una utilización promedio de 0.67%, la diferencia es muy pequeña. El uso de recursos del método RK4 del oscilador de Dos-Alas y del Jerk de Interruptor Único de 1.635% y 1.525% respectivamente, son los porcentajes más altos.

V. CONCLUSIONES

Las diferencias entre la simulación y la implementación del método de Euler Mejorado se deben a que la simulación realiza un cálculo de estados de una forma más lenta que la implementación física. Además, la implementación en VHDL aprovecha el paralelismo inherente de la tarjeta de FPGA, mientras el UART está a la espera de un mensaje de entrada, o mientras está a la espera de que el cliente descargue el mensaje de salida, el oscilador caótico se mantiene ejecutando. Así se garantiza que los estados utilizados para encriptar pixeles no son contiguos, mientras los píxeles si lo son, la simulación intenta emular este comportamiento, pero el hecho de “saltar” una gran cantidad de estados implica tener que generar aún más estados. En cambio, la diferencia entre la implementación y la simulación de Runge-Kutta de cuarto orden es menor, esto debido a que el paso utilizado ($h=0.1$) es diez veces más grande que el utilizado por el método de Euler Mejorado ($h=0.01$), por lo que los estados contiguos pueden presentar una mayor diferencia entre ellos. Con lo anterior se demuestra la factibilidad de implementar un sistema de encriptación digital en FPGA o inclusive en ASIC debido a que ninguno de los casos presentados utilizo más del 3% de algún tipo recurso lógico de la FPGA.

AGRADECIMIENTOS

Los autores agradecen al Instituto Politécnico Nacional por su apoyo y financiamiento provisto a través del proyecto “SIP-20210345”.

REFERENCES

- [1] S. Cang, A. Wu, Z. Wang, and Z. Chen, “Four-dimensional autonomous dynamical systems with conservative flows: two-case study,” *Nonlinear Dynamics*, vol. 89, no. 4, pp. 2495–2508, 2017.
- [2] H. Poincare, B. D. Popp, “The three-body problem and the equations of dynamics: Poincare’s foundational work on dynamical systems theory”, Primera edición. Cham, Switzerland: Springer, 2017.
- [3] P. J. Obeso-Rodelo, “Diseño e implementación en un fpga de oscilador caótico para aplicaciones en sistemas de seguridad”, Tesis de Maestría en Ciencias en Sistemas Digitales, CITEDI-IPN, Tijuana, B. C., México, 2015
- [4] H. M. Yassin et al., “Speech Encryption on FPGA Using a Chaotic Generator and S-Box Table” 2019 Fourth International Conference on Advances in Computational Tools for Engineering Applications (ACTEA), Beirut, Lebanon, 3 de Julio 2019, pp. 1-4.
- [5] S. Li, G. Chen, and X. Mou, “On the dynamical degradation of digital piecewise linear chaotic maps” *International Journal of Bifurcation and Chaos*, vol. 15, no. 10, pp. 3119–3151, 2005.
- [6] T. Bonny and A. S. Elwakil, “FPGA Realizations of High-Speed Switching-Type Chaotic Oscillators Using Compact VHDL Codes,” *non-linear dynamics*, vol. 93, no. 2, pp. 819–833, 2018.
- [7] Strogatz, S. Goldenfeld, N. “Sync: The Emerging Science of Spontaneous Order”, *Physics Today*, pp. 59-60, 2004.
- [8] H. M. Oliveira and L. V. Melo, “Huygens synchronization of two clocks,” *Scientific Reports*, vol. 5, no. 1, 2015.
- [9] H. Sira, C. Cruz. “Synchronization of Chaotic Systems: A Generalized Hamiltonian Systems Approach”, *International Journal of Bifurcation and Chaos*, Chicago, IL, USA, 28 de Junio 2000, pp. 764-773 vol.2.
- [10] L.J. Pei and S.H. Liu. Application of generalized Hamiltonian systems to chaotic synchronization. *Nonlinear Dynamics and Systems Theory*; 2009. 9(4): 415-432.

Implementación en Tarjeta de FPGA de Dos Sistemas Caóticos Conservativos Hamiltonianos

Jose Venegas Torres
and Gilberto Enrico Vazquez Alcaraz
Tecnológico Nacional de México
Instituto Tecnológico de Tijuana
Tijuana, Baja California, México, C. P. 22430
Email: jose.venegas18@tectijuana.edu.mx
gilberto.vazquez@tectijuana.edu.mx

Jose Cruz Nuñez Perez,
Miguel Angel Estudillo Valdez
and Andres Calvillo Tellez
Instituto Politécnico Nacional
IPN-CITEDI
Tijuana, Baja California, México, C. P. 22435
Email: jnunez@ipn.mx, {mestudillo,calvillo}@citedi.mx

Resumen—En este trabajo se presenta, por primera vez en la literatura, la implementación de dos sistemas caóticos conservativos hamiltonianos en una tarjeta de FPGA, utilizando dos métodos numéricos diferentes para resolver las ecuaciones diferenciales: el Euler mejorado y el Runge-Kutta de cuarto orden. Los osciladores caóticos que se implementan son dos casos de sistemas caóticos conservativos hamiltonianos de cuatro dimensiones, los cuales son simulados mediante el software MATLAB obteniendo sus cuatro diferentes series de tiempo. Posteriormente los dos casos son implementados digitalmente en una tarjeta de FPGA mediante la herramienta MATLAB/Simulink en conjunto con el software Vivado para el diseño digital en código en VHDL de los módulos. Por último se comparan ambos la simulación en Matlab con la emulación digital y los recursos lógicos utilizados. Las aportaciones originales de este trabajo son una plataforma digital para el diseño de sistemas caóticos capaz de desplegar la emulación digital de dos casos conservativos hamiltonianos de manera analógica, con la flexibilidad de cambiar sus condiciones iniciales y parámetros mediante software.

Palabras clave—Caos, Conservativo hamiltoniano, Euler, Matlab, Runge-Kutta, VHDL

I. INTRODUCCIÓN

En la actualidad las telecomunicaciones son parte fundamental de la vida diaria, ya que mantienen conectadas las redes de servicios digitales en el mundo. Estas redes de datos son capaces de transportar todo tipo de información, desde datos de dominio público hasta información privada. Por lo que la seguridad en los sistemas informáticos es parte esencial en las telecomunicaciones, debido a ello se han creado diversos sistemas de encriptación de datos para protegerlos de posible amenazas de robo y piratería. Sin embargo, al igual que los sistemas de seguridad han avanzado también los sistemas de piratearía lo han hecho, complicando la protección de la información.

Es aquí donde los sistemas caóticos pueden representar la nueva generación de sistemas de seguridad, esto debido a su imprevisibilidad. Los sistemas caóticos convencionales han demostrado altas capacidades para encriptar información, pero en los últimos años se han descubierto nuevos sistemas caóticos conservativos, estos son más complejos que los anteriores y tienen una característica única de pseudo-

aleatoriedad que los hace aún más difíciles de descifrar que los anteriores.

El estudio matemático del caos se remonta al siglo pasado cuando Edward Lorenz en 1963 describió una familia de tres parámetros de ecuaciones diferenciales ordinarias tridimensionales que parecía que al integrarse numéricamente en un ordenador, tenían soluciones extremadamente complicadas [1]. Después del descubrimiento de los sistemas caóticos como posible aproximación al modelo meteorológico en el planeta propuesto por Lorenz, los investigadores de todo el mundo buscaron nuevos sistemas y aplicaciones para los sistemas caóticos. Entre las principales aportaciones se descubrieron a los sistemas caóticos conservativos [2] los cuales, a diferencia de los sistemas caóticos convencionales, no cuentan con atractores generales. En [3] se propuso un nuevo método para construir sistemas caóticos conservativos de alta dimensión.

En tiempos más recientes se ha estudiado la mecánica de fluidos y cuerpos rígidos utilizando sistemas caóticos conservativos, como en [4] donde se generalizan cuatro ecuaciones de Euler para cuatro cuerpos sub-rígidos al extender la dimensión del espacio de 3D a 4D. Para esto se propusieron seis ecuaciones expresadas con el método de Euler en 4D y con ecuaciones sub-Euler con dos ejes en común. Al final se descubrió que al darle un límite al sistema, las orbitas del sistema caótico conservativo satisfacen el comportamiento deseado.

Por lo anterior se puede afirmar que los sistemas caóticos conservativos hamiltonianos (HCCS) son más complejos que los sistemas caóticos disipativos, por lo que su modelado e implementación es más complicada. Sin embargo, lo anterior acarrea mayores beneficios como se observa en [2] y [5], siendo capaces de emular las condiciones de energía y calor de objetos rígidos y fluidos. Además debido a su complejidad son mejores para la encriptación de información en medios digitales teniendo un grado alto de encriptación. En la actualidad se ha demostrado que los sistemas caóticos

se pueden utilizar para la encriptación de información [6] y [7] generando llaves mediante mapas caóticos, siendo capaces de encriptar información con coeficientes de correlación, entre la información original y la encriptada, menores al 1%. Algunos de estos sistemas de encriptado se han implementado en tarjetas FPGA con buenos resultados.

Este artículo está organizado de la siguiente manera: en la sección II se describen los sistemas caóticos conservativos hamiltonianos que se utilizaron para esta investigación, se introducen los dos casos: tipo I y tipo II. En la sección III se muestran los resultados de las simulaciones realizadas en Matlab, así como la descripción del comportamiento de los dos casos usando los métodos numéricos de Euler y Runge Kutta de cuarto orden. La sección IV presenta los resultados de implementación en una tarjeta FPGA usando VHDL, se describe además el diseño de una interfaz grafica que permite desplegar gráficamente el comportamiento de los dos casos conservativos hamiltonianos, con la flexibilidad de cambiar sus condiciones iniciales y parametros. Finalmente, las conclusiones se presentan en la sección V.

II. MARCO TEÓRICO

A. Sistemas Caóticos Conservativos Hamiltonianos

Los HCCS no cuentan con atractores generales a diferencia de los sistemas caóticos convencionales. La metodología para construir sistemas caóticos conservativos de alta dimensión, por ejemplo, sistemas de 4 y 5 dimensiones se describe en [3]. Se utilizó un sistema de 4D con variables de estado (x, y, z, w) , a partir del cual se construyó un sistema caótico conservativo generalizado, para ello se toman seis funciones de dos variables expresadas como (1):

$$F = f_1(z, w), f_2(y, w), f_3(y, z), f_4(x, w), f_5(x, z), f_6(x, y) \quad (1)$$

De esta serie de ecuaciones se derivaron dos sistemas caóticos conservativos Hamiltonianos diferentes, los cuales fueron denominados Tipo 1 y Tipo 2. Ambos sistemas cumplen con la conservación de energía de volumen, parámetros requeridos para que sean considerados sistemas conservativos.

El HCCS Tipo 1 tiene la cualidad de que la energía Hamiltoniana es conservativa mientras que la energía de casimir no lo es y está dada por el sistema de ecuaciones (2),

$$\begin{aligned} \dot{x} &= hyw + cdz \\ \dot{y} &= hzw - 0.5hxw \\ \dot{z} &= -hyw - cbx \\ \dot{w} &= -0.5hxy \end{aligned}$$

donde los coeficientes h, b, c, d son los parámetros del HCCS Tipo 1.

El HCCS Tipo 2 es ligeramente diferente al primero, pero en este caso la energía Hamiltoniana es la que varía a lo largo

del tiempo. El Tipo 2 está dado por el sistema de ecuaciones (2),

$$\begin{aligned} \dot{x} &= hyw + gz \\ \dot{y} &= azw + exw \\ \dot{z} &= -ayw - gx \\ \dot{w} &= -(h + e)xy \end{aligned}$$

donde los coeficientes h, a, e, g son los parámetros del HCCS Tipo 2.

B. Métodos Numéricos

Euler mejorado: El primer método utilizado para resolver los sistemas será el de Euler mejorado. El método tiene la misma estructura básica, se define el sistema de ecuaciones de cada uno de los sistemas, después se propone un tamaño de paso h necesario para el calculo de ambos métodos y por último un ciclo *for* que va desde 0 hasta n , donde n representa la cantidad de muestras.

Runge-Kutta de cuarto orden: El procedimiento para resolver los sistemas de ecuaciones por el método de Runge Kutta de cuarto orden es igual al de Euler mejorado, solo cambian las ecuaciones utilizadas en el ciclo *for* por las de Runge Kutta de cuarto orden. El resto es idéntico, se define el sistema de ecuaciones dependiendo del sistema a resolver, después los valores de las contantes y de las condiciones iniciales, estos son los mismos que se utilizaron en el método de Euler mejorado, y por último se aplica un ciclo *for* para calcular los puntos que tomará el sistema de ecuaciones a lo largo del tiempo.

III. SIMULACIÓN

En esta sección se presentan los resultados de simulación de los dos HCCS Tipo 1 y 2, tanto en Matlab como en Vivado y la implementación de los sistemas en simulink para su emulación. A fin de obtener la solución de los sistema caóticos se utilizaron los métodos numéricos Euler mejorado y Runge kutta de cuarto orden.

A. Euler Mejorado

1) *Sistema caótico conservativo Hamiltoniano tipo 1:*
El HCCS Tipo 1 utiliza el sistema de ecuaciones (2) y se simula empleando los valores para las constantes $(h, d, b, c) = (-1, 2.5, 2, 0.2)$ y las condiciones iniciales $(x_0, y_0, z_0, w_0) = (0.5, 0.5, 1, 1)$. En la Figura 1a se muestran los resultados obtenidos en los planos 2D: $x - y, x - z, y - w$. Mientras que la Figura 1b muestra los resultados en los planos 3D: $x - y - w, x - y - z$.

2) Sistema caótico conservativo Hamiltoniano tipo 2:

El HCCS Tipo 2 utiliza el sistema de ecuaciones (3) y las constantes $(h, g, a, e) = (0.5, 10, 6, 5.5)$ y las condiciones iniciales $(x_0, y_0, z_0, w_0) = (1, 1, 1, 1)$. En la Figura 2a se muestran los resultados obtenidos en los planos 2D: $x - y, x - z, y - w$. Mientras que la Figura 2b muestra los resultados en los planos 3D: $x - y - w, x - y - z$.

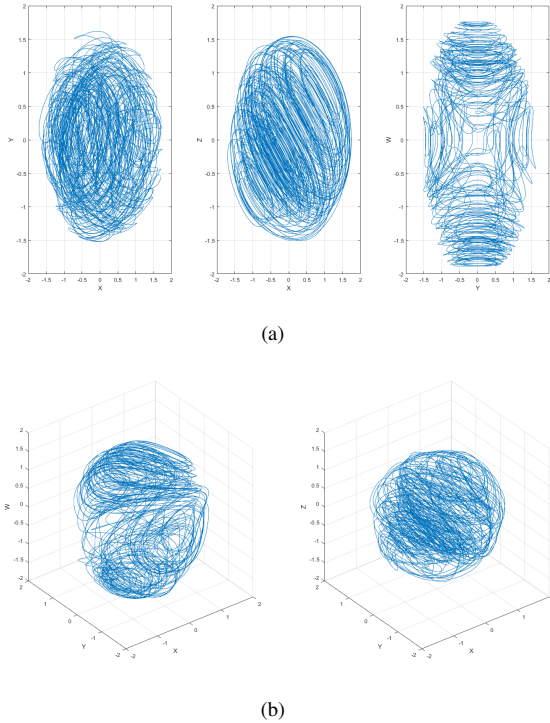


Fig. 1. Gráficas del HCCS Tipo 1 con Euler Mejorado, a) En los planos 2D $x - y, x - z, y - w$, b) En los planos 3D $x - y - w, x - y - z$

El HCCS Tipo 2 puede tomar otros valores en sus condiciones iniciales y constantes los cuales son los siguientes: $(h, a, e, g) = (-25, -2, 1, 0.7)$ y las condiciones iniciales pueden variar entre cuatro diferentes posibilidades: $(x_0, y_0, z_0, w_0) = (\pm 0.0918, y^*, \pm 0.00469, \pm 0.00290)$ donde $y^* = 0.05, 0.08$, con estos valores se generan las gráficas mostradas en la Figura 3. Solo se presenta una parte del sistema en las gráficas para que sea más sencillo identificar las diferencias entre cada gráfica.

B. Runge-Kutta de Cuarto Orden

1) Sistema caótico conservativo Hamiltoniano Tipo 1:

El HCCS Tipo 1 utiliza el sistema de ecuaciones (2) y utilizando las constantes $(h, d, b, c) = (-1, 2.5, 2, 0.2)$ y las condiciones iniciales $(x_0, y_0, z_0, w_0) = (0.5, 0.5, 1, 1)$. En la Figura 4a se ilustran los resultados obtenidos en los planos 2D: $x - y, x - z, y - w$. Además la Figura 4b muestra los resultados en los planos 3D: $x - y - w, x - y - z$.

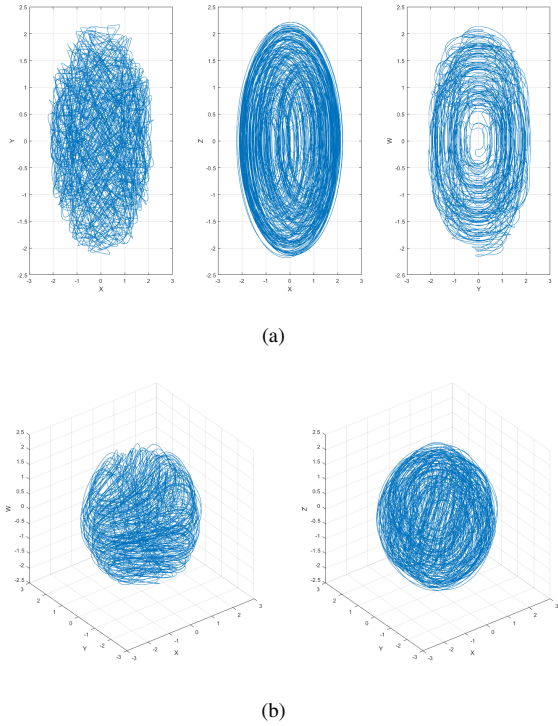


Fig. 2. Gráficas del HCCS Tipo 2 con Euler Mejorado, a) En los planos 2D $x - y, x - z, y - w$, b) En los planos 3D $x - y - w, x - y - z$

2) Sistema caótico conservativo Hamiltoniano tipo 2: El HCCS Tipo 2 utiliza el sistema de ecuaciones (3) y emplea las constantes $(h, g, a, e) = (0.5, 10, 6, 5.5)$ y las condiciones iniciales $(x_0, y_0, z_0, w_0) = (1, 1, 1, 1)$ se obtienen las gráficas mostradas en la Figura 5.

Tal como se mencionó en el caso de Euler Mejorado en la subsección III-A2, el HCCS Tipo 2 puede tomar otros valores en sus condiciones iniciales y constantes. Usando los mismos que en la subsección III-A2 se generan las gráficas mostradas en la Figura 6.

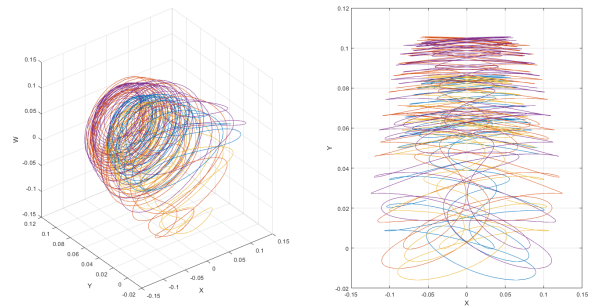


Fig. 3. Grafica del sistema tipo 2 segunda representación

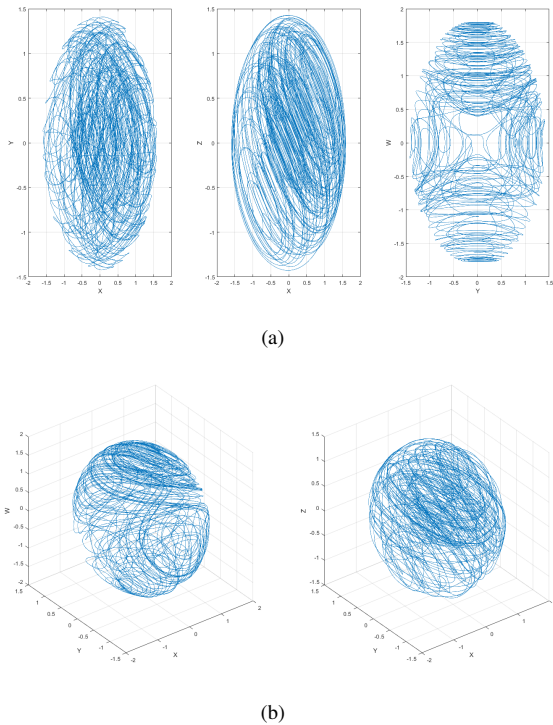


Fig. 4. Gráficas del HCCS Tipo 1 con RK4, a) En los planos 2D $x - y, x - z, y - w$, b) En los planos 3D $x - y - w, x - y - z$

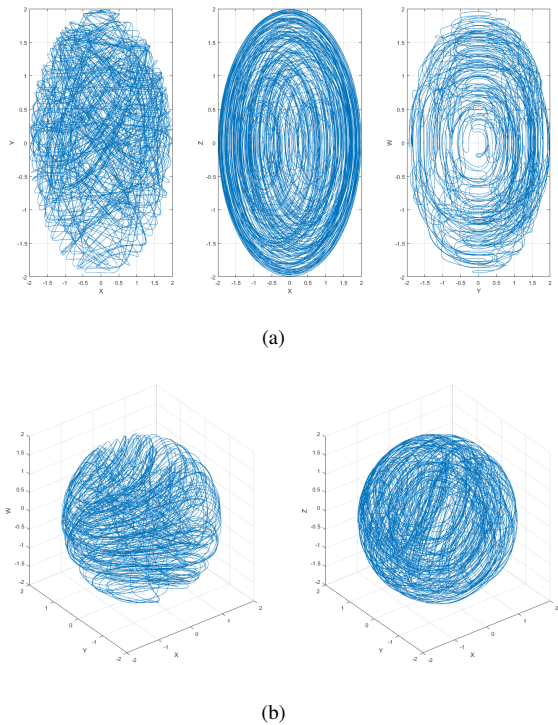


Fig. 5. Gráficas del HCCS Tipo 2 con RK4, a) En los planos 2D $x - y, x - z, y - w$, b) En los planos 3D $x - y - w, x - y - z$

IV. IMPLEMENTATION EN CÓDIGO VHDL

En la implementación del código realizado en VHDL se utilizó el System Generator, la cual es una extensión de Vivado del fabricante Xilinx que permite comunicarse con MATLAB. Esto agrega algunas funciones nuevas a la extensión Simulink de MATLAB, las cuales permiten utilizar bloques para convertir señales analógicas a digitales, entre otras cosas. Entre los bloques que proporciona System Generator se encuentra uno que permite usar un código diseñado en Vivado llamado “Black Box”, para esto solo se requiere introducir la dirección donde se encuentra el archivo generado por Vivado. Adicionalmente se tiene que configurar la relación de tiempo entre ambos programas, por defecto este valor se encuentra en 10ns de Vivado, equivalente a 1 segundo de Simulink. Debido a esto fue necesario agregar un retraso (delay) en el código de VHDL de 100us, de lo contrario Simulink no era capaz de recibir todos los datos.

Una vez que ha sido configurado el bloque “Black Box” se adapta al número de salidas definidas en el código de VHDL. Para los dos Tipos de HCCS fueron cuatro salidas, una por cada variable. Este bloque no tiene entradas por definición del código realizado y tampoco requiere de un temporizador, CLK, para funcionar. En MATLAB-Simulink cada una de las salidas fue conectada a un bloque “SCOPE” para representar los datos en forma de gráfica. A fin de mejorar la calidad en la presentación de los resultados fue necesario exportar los

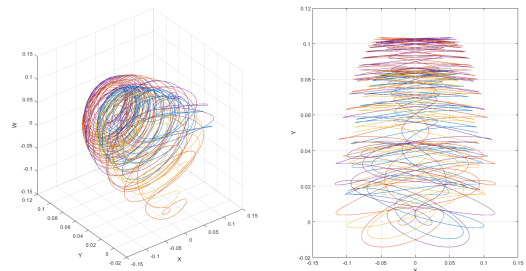


Fig. 6. Gráfica del sistema tipo 2 segunda representación

datos al espacio de trabajo de MATLAB para generar gráficas en diferentes planos.

Se realizaron dos códigos, uno para cada tipo de HCCS, los cuales tienen una estructura similar, empezando por la descripción de las variables que se utilizan. Estas variables son trabajadas con el tipo de dato STD LOGIC VECTOR al cual se le designaron 8 bits para representar los valores que se utilizan como salidas del sistema, teniendo un total de cuatro variables, (x, y, z, w) , por ende el sistema cuenta con cuatro salidas. Debido a que se realizan operaciones aritméticas para calcular los puntos que toman las ecuaciones del sistema, fue necesario trabajar con tipo de dato REAL ya que este permite utilizar operaciones básicas. En este tipo de dato no es necesario definir un tamaño en bits para representar los

valores, pero tiene el inconveniente de no ser compatible con las salidas digitales de la tarjeta de FPGA. Por lo tanto se implementó una función para convertir los valores de REAL a STD LOGIC VECTOR y así poder mandar estos valores a las salidas del sistema.

Ambos códigos comparten todas las características anteriores, la diferencia entre los dos se encuentra en la definición de los sistemas de ecuaciones, condiciones iniciales y valores contantes de cada sistema.

En el diseño del diagrama de bloques en MATLAB-Simulink se utilizó la extensión System Generator de Xilinx Vivado, con esta extensión es posible utilizar algunos bloques para utilizar códigos hechos en Vivado directo en MATLAB-Simulink. La Figura 7 muestra el diagrama diseñado.

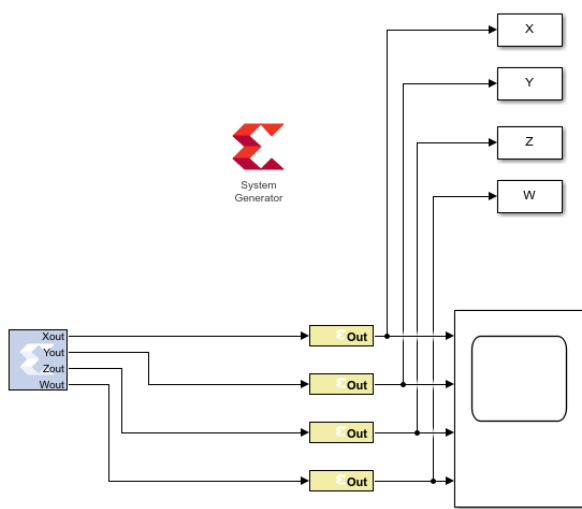


Fig. 7. Diagrama de implementación de los HCCS tipo I y II en una tarjeta FPGA Xilinx usando el System Generator

Por último se diseñó una interfaz gráfica con la ayuda de la herramienta GUIDE de Matlab, esta herramienta nos permite colocar elementos básicos, botones, tablas, listas, texto fijo, etc., y modificar su comportamiento mediante código base de Matlab.

La interfaz consta de 3 pantallas, la primera da información sobre los sistemas caóticos que se utilizaron, HCCS. Esta pantalla tiene dos botones para seleccionar entre los dos sistemas, Tipo 1 o Tipo 2. Una vez seleccionado el HCCS se despliega la segunda pantalla, en ella se selecciona el método numérico que se utilizará para resolver el sistema de ecuaciones diferenciales. Además de tener una serie de botones para graficar el sistema en sus diversos planos, un botón para graficar las series de tiempo, tres botones para graficar 3 planos diferentes en dos dimensiones y por ultimo dos botones para graficar dos planos en tres dimensiones. Una vez seleccionada

la gráfica se despliega la tercer y última pantalla, en esta solo se encuentran las gráficas del sistema y un botón para volver a la pantalla anterior.

V. RESULTADOS

En esta sección se describen los resultados obtenidos en la implementación de los dos tipos de HCCS en una tarjeta FPGA. En el HCCS Tipo 1 se configuró el bloque "Black Box" del diagrama de bloques de MATLAB-Simulink para que utilice el código VHDL diseñado, emulando así la implementación en una tarjeta de FPGA. En el HCCS Tipo 2 se realizó el mismo procedimiento anterior con el código VHDL diseñado. La Figura 8 muestra los resultados de las series de tiempo x, y, z, w , para los HCCS (a) Tipo 1 y (b) Tipo 2, respectivamente.

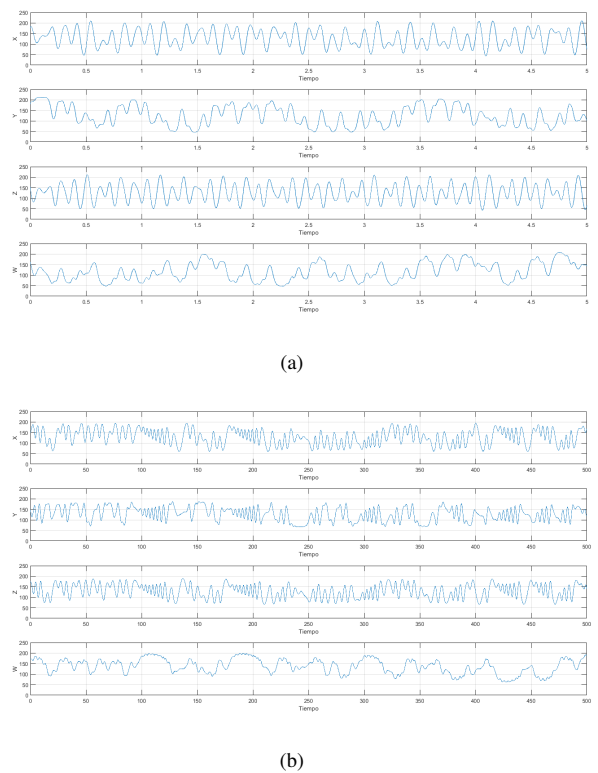


Fig. 8. Series de tiempo x, y, z, w del HCCSs, a) Tipo I, b) Tipo II

La Figura 9 ilustra las ventanas 1 y 2 de la interfaz grafica de usuario, donde el operador puede configurar y seleccionar los resultados de implementación en FPGA a visualizar: el caso de HCCS (Tipo 1 ó Tipo 2) y los planos a desplegar (en 2D ó 3D).

La Figura 10 ilustra los resultados del HCCS en el plano 2D ($x - z$) y el plano 3D ($x - y - z$) para el HCCS Tipo 1 y Tipo 2, respectivamente.

En la implementación en FPGA se utilizó la tarjeta de FPGA Kintex-7 de Xilinx, XC7K70TFBV676-1. La Tabla I

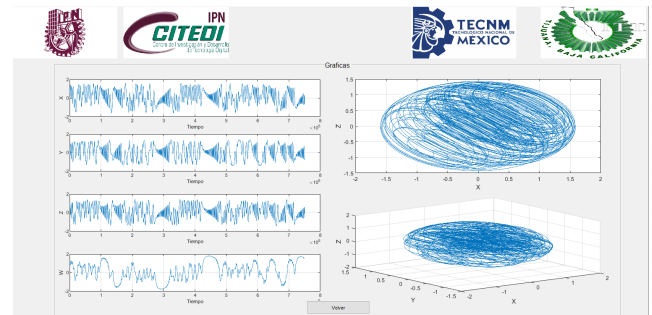


(a)

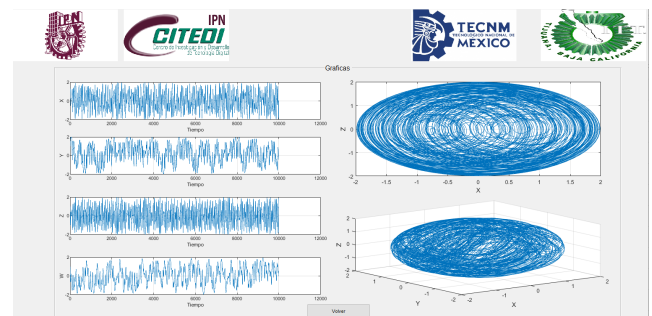


(b)

Fig. 9. Interfaz gráfica diseñada, a) Ventana 1, selección de tipo de HCCS, b) Ventana 2, selección de plano a visualizar



(a)



(b)

Fig. 10. Gráficas del HCCS en el plano 2D ($x-z$) y el plano 3D ($x-y-z$), a) Tipo 1, b) Tipo 2

muestra la comparación del empleo de recursos lógicos en la implementación en FPGA de los HCCS Tipo 1 y Tipo 2 con el método numérico de Euler Mejorado. Además la Tabla II muestra los recursos lógicos utilizados en la implementación en FPGA de los HCCS Tipo 1 y Tipo 2, usando el método de Runge-Kutta de cuarto orden.

El análisis de ambas tablas establece al HCCS Tipo 2 por Euler Mejorado como la mejor opción en el uso de recursos con una utilización promedio de 26.67% de la tarjeta Kintex-7. Si bien, en su homólogo HCCS Tipo 1 el que menos recursos lógicos utiliza con una utilización promedio de 35%, la diferencia es por tanto considerable. El uso de recursos del método de Runge-Kutta de cuarto orden de los HCCS Tipo 1 y Tipo 2 fue de 53.3% y 70% respectivamente, el doble con respecto al de Euler Mejorado.

VI. CONCLUSIÓN

En este artículo se llevó a cabo un estudio bibliográfico del estado del arte sobre osciladores caóticos disipativos y conservativos. Se revisaron los métodos numéricos y su importancia para resolver cualquier tipo de sistema caótico, entre estos métodos destacan el método de Euler mejorado y el método de Runge Kutta de cuarto orden. Se demostró el diseño de dos tipos diferentes de sistemas caóticos conservativos hamiltonianos, de los cuales primeramente se presentaron los resultados de las simulaciones en el software de MATLAB. Ambos fueron resueltos tanto por el método de Euler mejorado como por el de Runge Kutta de cuarto orden. Al ser sistemas

TABLA I
RECURSOS LÓGICOS UTILIZADOS POR LOS HCCS TIPO 1 Y 2, CON EL MÉTODO DE EULER MEJORADO.

HCCS Tipo 1			
Recurso	Utilizado	Disponible	Porcentaje
I/O pins	128	300	42.67%
DSPs	84	240	35%
Memory LUTs	5279	41000	12.88%
LUTs	0	82000	0%
HCCS Tipo 2			
Recurso	Utilizado	Disponible	Porcentaje
I/O pins	129	300	43%
DSPs	64	240	26.67%
Memory LUTs	5869	41000	14.31%
LUTs	256	82000	0.31%

TABLA II
RECURSOS LÓGICOS UTILIZADOS POR LOS HCCS TIPO 1 Y 2, CON EL MÉTODO DE RUNGE-KUTTA CUARTO ORDEN.

HCCS Tipo 1			
Recurso	Utilizado	Disponible	Porcentaje
I/O pins	128	300	42.67%
DSPs	128	240	53.33%
Memory LUTs	14786	41000	36.06%
LUTs	0	82000	0%
HCCS Tipo 2			
Recurso	Utilizado	Disponible	Porcentaje
I/O pins	128	300	42.67%
DSPs	168	240	70%
Memory LUTs	13235	41000	32.28%
LUTs	0	82000	0%

tan complejos se identificó un par de diferencias a simple vista entre ambos métodos, resultando más preciso el método de Runge Kutta de cuarto orden. También se mostró el procedimiento que se siguió para implementar los sistemas en código VHDL. Finalmente se mostraron los resultados de la implementación del código VHDL de ambos sistemas en un ambiente de emulación en Matlab-Simulink, utilizando algunas de las herramientas que provee Vivado para comunicarse con MATLAB. De esta manera se puede utilizar directamente el código diseñado en Vivado para la implementación en MATLAB-Simulink, comprobando con esto que es posible implementar los dos sistemas en una tarjeta de FPGA.

AGRADECIMIENTOS

Los autores agradecen al Instituto Politécnico Nacional por su apoyo y financiamiento provisto a través del proyecto “SIP-20210345”.

REFERENCES

- [1] S.-K. Yang, C.-L. Chen, and H.-T. Yau, Control of chaos in Lorenz system. *Chaos, Solitons Fractals*, Vol. 13, No. 4, 767–780, 2002.
- [2] J.-Rose, A.-El Gamal, A.-Sangiovanni-Vincentelli, Architecture of field-programmable gate arrays. *Proceedings of the IEEE*, Vol. 81, No. 7, 1013–1029, 1993.
- [3] G. Qi, Modelings and mechanism analysis underlying both the 4D Euler equations and Hamiltonian conservative chaotic systems. *Nonlinear Dynamics*, Vol. 95, 2063–2077, 2019.
- [4] E. Dong, M. Yuan, S. Du, Chen, Z. A new class of Hamiltonian conservative chaotic systems with multistability and design of pseudo-random number generator. *Applied Mathematical Modelling*, Vol. 73, 41-71, 2019.
- [5] E. Tlelo-Cuautle, J.J. Rangel-Magdaleno, A. D. Pano-Azucena, P. J. Obeso-Rodelo, J.C. Nuñez-Perez, FPGA realization of multi-scroll chaotic oscillators, *Communications in Nonlinear Science and Numerical Simulation*, Vol. 27, No. (1-3), 66-80, 2015.
- [6] F. Yang, J. Mou, J. Liu, C. Ma, H. Yan, Characteristic analysis of the fractional-order hyperchaotic complex system and its image encryption application. *Signal Processing*, Vol. 169, No. 107373, 2020.
- [7] F. Yang, J. Mou, C. Ma, Y. Cao, “Dynamic analysis of an improper fractional-order laser chaotic system and its image encryption application”. *Optics and Lasers in Engineering*, Vol. 129, No. 106031, 2020.

Detección de Microaneurismas en Retinografía Mediante el Uso de Operaciones Morfológicas

Yeredith Giovanna Mora M., Volodymyr Ponomaryov, Rogelio Reyes R., Clara Cruz R.

Instituto Politécnico Nacional, ESIME Unidad Culhuacán

Av. Santa Ana, No.1000, Col. San Francisco Culhuacán, CP. 04430, México, Ciudad de México.

Tel. 57296000, Ext. 73264 Fax (55) 56562058

e-mail: ymoram1500@alumno.ipn.mx, vponomar@ipn.mx, rreyesre@ipn.mx, ccruzra@ipn.mx

Resumen. - La retinopatía diabética (RD) es una enfermedad crónica progresiva que se asocia al aumento de azúcar en sangre (hiperglucemia), la cual es la principal causa de pérdida de visión. Por esta razón la detección temprana en la progresión de la RD es de gran importancia, debido a que es un primer paso en la prevención de la evolución de esta enfermedad, además de que el especialista puede dar un tratamiento adecuado. Este trabajo consta de tres etapas principales para la detección y clasificación de microaneurismas (MA) en retinografía, la primera etapa atenúa los efectos del ruido preservando la información de los MA candidatos, en la segunda etapa se extraen los MA candidatos y la tercera etapa consta de extraer las características de los MA, para finalmente clasificarlas utilizando el algoritmo de aprendizaje supervisado de k-vecinos más próximos (KNN). Los experimentos realizados demostraron que el sistema propuesto es capaz de separar correctamente los MA del fondo del ojo, con una exactitud del 91.62% y una precisión del 84.6%.

Palabras Clave: Retinopatía diabética, detección de microaneurismas, clasificación de microaneurismas, extracción de características.

I. INTRODUCCIÓN

En un estudio poblacional en el sur de Wisconsin, EE. UU. determinaron que la retinopatía diabética (RD) es una complicación microvascular de la diabetes y la principal causa de ceguera y pérdida de visión [1], en México se estima que más de 3 millones 600 mil pacientes con diabetes presentan algún grado de daño visual por RD, enfermedad que se ha convertido en la primera causa de ceguera en adultos en etapa productiva [2]. Aquellas personas que padecen retinopatía y no tienen un control glucémico adecuado son más propensas a tener retinopatía diabética, los vasos sanguíneos llegan a bloquearse debido al alto nivel de azúcar en sangre, por lo que los ojos intentan cultivar nuevos vasos sanguíneos y al no ser desarrollados adecuadamente son propensos a fugas afectando el sistema visual [3].

La detección de microaneurismas (MA) es de gran importancia, debido a que son el primer signo clínicamente identificable de la retinopatía diabética no proliferativa, y esto puede presentar un primer paso para la prevención de la evolución a la fase proliferativa y así el especialista pueda dar un tratamiento adecuado. Los MA aparecen como puntos rojos circulares en la retina, la complejidad en su reconocimiento se debe a que pueden ubicarse en cualquier lugar de la retina, ya sea de forma aislada, en grupos, cerca de la vasculatura, alrededor de la macula o entre exudados por lo que su reconocimiento automático desde el punto de vista

computacional es complejo. Por otra parte, su contraste local es muy bajo en comparación con su fondo circundante y sus bordes no están bien definidos en la imagen. Además, los MA tienen características morfológicas e intensidad muy similares a otros signos de retinopatías y características anatómicas como hemorragias, vasos delgados y ruido de fondo de la retina [4].

Debido a lo anterior se propone un sistema computacional para la detección de MA en imágenes digitales de fondo de ojo (retinografía) mediante el procesamiento de imágenes médicas empleando operaciones morfológicas.

II. ANTECEDENTES

Arati Manjramkar et al. [5] proponen un algoritmo que consta de tres pasos, inicialmente se realiza el preprocesamiento de la imagen, en segundo lugar, se segmentan los MA candidatos y por último se extraen las características de estos candidatos y se reconocen los MA verdaderos donde utilizan un sistema experto basado en reglas. El algoritmo clasifica erróneamente algunos MA debido al ruido producido en la etapa de segmentación, sus resultados obtenidos dieron una sensibilidad del 80.6%, especificidad del 97.50% y exactitud del 95.95%.

Rukhmini Roy et al. [6] proponen un algoritmo que consta de dos etapas, la primera etapa comprende de un preprocesamiento y análisis fractal de la estructura vascular retiniana, la segunda etapa detecta los MA a medida que la imagen anormal pasa por la detección de bordes y la reconstrucción morfológica. Los resultados que obtuvieron dieron una sensibilidad del 89.5% y una especificidad del 82.1% en la detección de MA.

Meindert Niemeijer et al. [7] proponen un algoritmo que combina trabajos de Spencer et al. [8] y Frame et al. [9] proponiendo nuevas contribuciones. La primera contribución es la detección de lesiones rojas basado en clasificación de píxeles, la cual separa la vasculatura y las lesiones rojas del fondo de la imagen. La segunda contribución agrega más características a las propuestas por Spencer-Frame, clasificándolo mediante k-vecinos más próximos. El algoritmo elimina la vasculatura sin tocar los posibles MA candidatos removiendo lesiones brillantes, sin embargo, los MA que están cercanos a la vasculatura son confundidos fácilmente como parte de esta, los resultados obtenidos en este método alcanzaron una sensibilidad del 100% y una especificidad del 82%. Por su parte, Frame et al. [9] proponen un total de 13 características diferentes de objetos candidatos para usar con el sistema Spencer-Frame. Donde las características pueden ser divididas

en dos grupos: características de forma y características de intensidad de píxeles.

El algoritmo propuesto realiza durante el preprocesamiento una corrección de sombras, lo cual disminuye significativamente el ruido en la imagen, sin embargo se preservan las estructuras oscuras y pequeñas, lo que permite que se realice una mejor segmentación, lo que implica una disminución considerable en los FN en comparación con el método de Manjaramkar et al. [5] y Rukhmini Roy et al. [6], por otra parte, también se logran preservar los MA cercanos a la vasculatura lo que genera mejores resultados en comparación con Meindert Niemeijer et al. [7], sin embargo, podemos considerar como una desventaja es que, en la etapa de extracción de los MA candidatos prevalecen regiones en donde se mantienen residuos de la vasculatura, los cuales, pueden ser confundidos por el sistema como MA verdaderos, lo que ocasiona un aumento en el número de FP.

III. SISTEMA PROPUESTO

El diagrama a bloques del sistema propuesto se muestra en la Fig. 1. Consta de tres etapas principales: preprocesamiento, extracción de MA candidatos y extracción de características. Finalmente la etapa del clasificador. Los detalles de estas etapas se dan a conocer en las secciones siguientes.

A. Etapa de preprocesamiento

Una imagen de fondo de ojo de tamaño 1500x1152 píxeles entra al bloque de preprocesamiento, y es redimensionada al estándar PAL de 576x720 píxeles representada como I_r .

De la imagen I_r se obtiene el canal verde I_g , el cual es seleccionado debido a que los MA tienen mayor contraste en el plano verde, por lo que la información del plano rojo y azul no se utiliza.

A la imagen I_g se le aplica un filtro gaussiano con tamaño de ventana 3 y varianza 1, para mejorar las estructuras oscuras y pequeñas, dando como resultado la imagen I_{gg} . Para la prevención de los falsos positivos se realizó un algoritmo de corrección de sombra en la imagen I_{gg} el cual se describe a continuación.

- 1) Primero se estima la imagen de fondo I_{bg} aplicando un filtro de mediana de 35x35 a I_{gg} .
- 2) Se resta I_{bg} de I_{gg} para obtener I_{sc} . Todos aquellos píxeles en I_{sc} que tengan valores positivos en I_{gg} , corresponden a los píxeles que tienen valores de intensidad más altos.
- 3) Por último, los píxeles en I_{gg} indicados por I_{sc} son reemplazados por los valores de sus píxeles correspondientes en I_{bg} resultando en una imagen I_{pp} .

B. Extracción de MA candidatos

Esta etapa consta de tres partes como se muestra en la Fig. 1, la primera parte realiza la detección de bordes mediante el algoritmo de Canny, en la segunda parte se lleva a cabo el relleno de las imágenes binarias y por último se elimina la vasculatura.

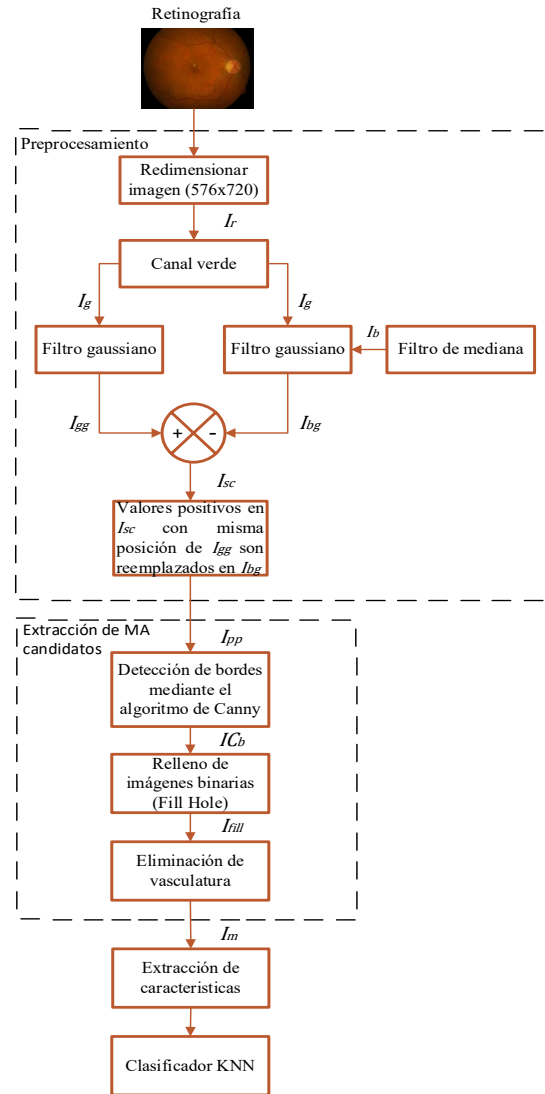


Fig. 1. Diagrama general del sistema propuesto.

a) *Detección de bordes mediante el algoritmo de Canny:* La detección de bordes consiste en tres pasos como se muestra en la Fig. 2.

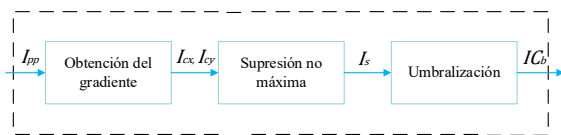


Fig. 2. Detección de bordes mediante el algoritmo de Canny.

El primer paso consiste en obtener el gradiente de la imagen I_{pp} , calculando la magnitud y la orientación del vector gradiente. Para este primer paso es necesario obtener la primera derivada la cual permite identificar un máximo local de la función de la imagen ante la presencia de un borde, en las coordenadas x e y . Para una función $f(x, y)$ al vector que apunta en la dirección de la variación máxima de $f(x, y)$ es denominado gradiente y es definida por (1), su magnitud por

(2) mide la razón del máximo cambio de intensidad en la ubicación (x_0, y_0) y su dirección definida en (3) es la del mayor aumento en la intensidad.

$$\nabla f(x, y) = \left[\begin{array}{c} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{array} \right], \quad (1)$$

$$Mag[\nabla f(x, y)] = \sqrt{\left(\frac{\partial f(x, y)}{\partial x}\right)^2 + \left(\frac{\partial f(x, y)}{\partial y}\right)^2}, \quad (2)$$

$$\theta = \arctag \frac{\frac{\partial f(x, y)}{\partial x}}{\frac{\partial f(x, y)}{\partial y}}. \quad (3)$$

El segundo paso se denomina supresión no máxima, esta se realiza adelgazando el ancho de los bordes, teniendo como resultado un borde de un píxel de ancho. Para realizar la supresión no máxima se toman las imágenes I_{cx} e I_{cy} para generar una imagen con bordes delgados tomando en cuenta cuatro direcciones (0° , 45° , 90° y 135°). Se observa si el valor de la magnitud es más pequeño que al menos uno de sus vecinos en la dirección del ángulo obtenido anteriormente, de ser así se le asigna un 0 al píxel, en caso contrario se le asigna el valor de la magnitud del gradiente.

El tercer paso es la umbralización de la imagen I_s , denominada histéresis de umbral, donde se aplica una función de histéresis (conservando sus propiedades) basada en dos umbrales (T_1, T_2).

b) Fill Hole: En la etapa de detección de bordes se obtuvo una imagen IC_b a la cual fue necesario eliminar la vasculatura para mantener solamente los MA candidatos, para esto es necesario rellenar los agujeros de la imagen mediante una operación morfológica conocida como dilatación. Primero se toma la imagen IC_b y se obtiene su imagen complemento, así mismo se define el elemento estructurante. Sucesivamente se toma como punto de partida el interior del objeto a rellenar y se dilata la imagen, se realiza este paso “n-veces” hasta llegar al borde, obteniendo una imagen I_{fill} .

c) Eliminación de vasculatura: Para la eliminación de vasculatura, a la imagen IC_b se le resta la imagen construida morfológicamente I_{fill} creando una imagen I_m . Al realizar este paso la imagen I_m queda con MA candidatos, así como fragmentos de vasos y poco ruido, los MA y el ruido tienen un área similar, sin embargo, su principal diferencia se encuentra en su forma, debido a que los MA tienen forma circular y el ruido forma alargado o irregular.

C. Extracción de características

Finalmente, para discriminar a los MA verdaderos se obtuvo un total de 6 características para cada MA candidato, los cuales se obtienen empleando (4)-(7).

- 1) Para el cálculo del área se utiliza (4)

$$\sum_{j \in \Omega} 1, \quad (4)$$

donde Ω es el conjunto de píxeles del candidato.

2) El perímetro p se calcula como si los píxeles delimitadores del candidato estuvieran representados por una cadena de 8 códigos. Cada enlace horizontal o vertical aporta 1 unidad y cada enlace diagonal proporciona $\sqrt{2}$ unidades al perímetro.

- 3) La circularidad se calcula con (5)

$$C = \frac{4\pi Area}{p^2}, \quad (5)$$

la cual ofrece una medida de redondez y suavidad de los límites del candidato.

- 4) El diámetro se calcula con (6)

$$D = \sqrt{\frac{4Area}{\pi}}. \quad (6)$$

5) La excentricidad, obtiene los valores de cada candidato y se encuentra en el rango de 0 a 1.

- 6) La relación de aspecto se calcula con (7)

$$s = \frac{l}{w}, \quad (7)$$

la cual es definida como la relación entre la longitud más larga del objeto y la anchura.

IV. PRUEBAS Y RESULTADOS

Para la realización de las pruebas se utilizó una base de datos de retinopatía diabética DIARETDB1 [10], la cual fue publicada por la universidad de Kuopio, Finlandia. Esta base de datos es un conjunto de imágenes de fondo de ojo RGB de alta resolución de tamaño de 1500×1152 píxeles, las cuales fueron reescaladas a un tamaño estándar PAL de 576×720 .

El desarrollo y las pruebas realizadas se hicieron en una laptop con procesador AMD A8 y una velocidad de 2GHz, con memoria RAM de 4GB y sistema operativo Windows 10, el sistema fue implementado en Matlab 2016b.

En la Fig. 3 se muestran los resultados de la etapa de preprocesamiento, donde se puede observar en la Fig. 3(b) el canal verde de la retinografía, en la Fig. 3(c) se muestra la imagen I_g después de aplicar el filtro gaussiano obteniéndose I_{gg} , en la Figura 3(d) se muestra la imagen I_{gg} después de aplicar el filtro de mediana dando como resultado I_{bg} , en la Fig. 3(e) se muestra la imagen de corrección de sombras I_{sc} y finalmente en la Fig. 3(f) se muestra la imagen preprocesada I_{pp} .

En la Fig. 4 se muestra la etapa de detección de bordes por el algoritmo de Canny, donde en la Fig. 4(a-b) muestra las imágenes del cálculo de la gradiente de la imagen I_{pp} , en la Fig. 4(c) se muestra la imagen de la supresión no máxima, la cual muestra un adelgazamiento en los bordes (I_s) y finalmente en la Fig. 4(d) se muestra la imagen de la umbralización denominada histéresis del umbral, la cual conecta los bordes de la imagen (IC_b).

En la Fig. 5(a) se muestra la imagen construida morfológicamente denominada I_{fill} , en la Fig. 5(b) se muestra la imagen con la vasculatura ya eliminada I_m .

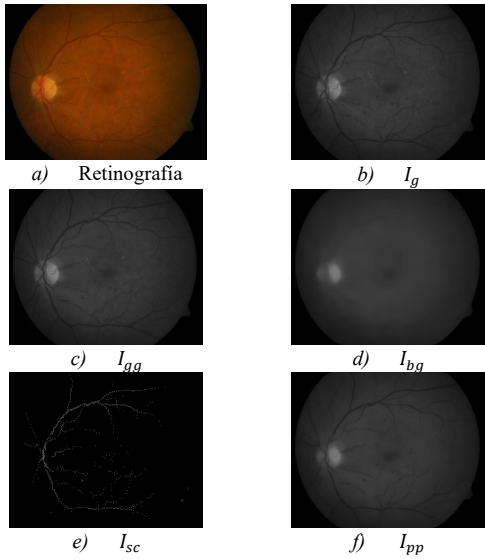


Fig. 3. Imágenes resultantes de la etapa de preprocesamiento.



Fig. 6. Imagen de MA candidatos.

En la Tabla 1 se muestran algunos resultados obtenidos del sistema desarrollado, donde se observa la imagen de fondo de ojo, así como los resultados del preprocesamiento realizado, detección de bordes y MA candidatos; en donde el tiempo promedio de ejecución para cada imagen es de 6.52 segundos, cabe mencionar que este tiempo también depende del número de MA encontradas en la imagen.

TABLA I. RESULTADOS OBTENIDOS CON EL SISTEMA PROPUESTO

Retinografía	Pre-procesamiento	Detección de bordes	MA candidatos

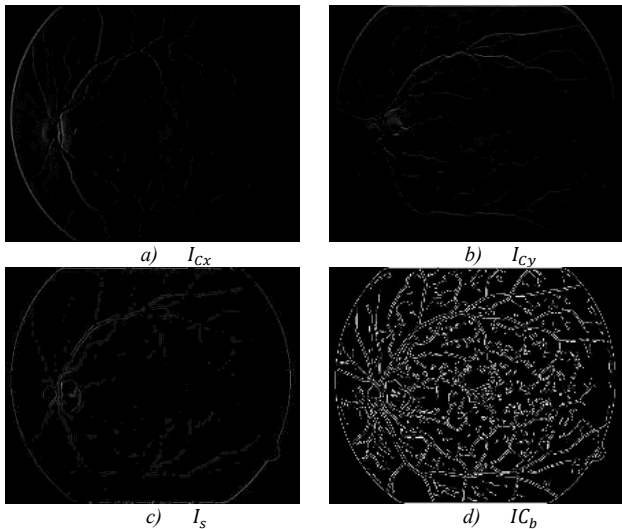


Fig. 4. Imágenes resultantes de la etapa de detección de bordes.

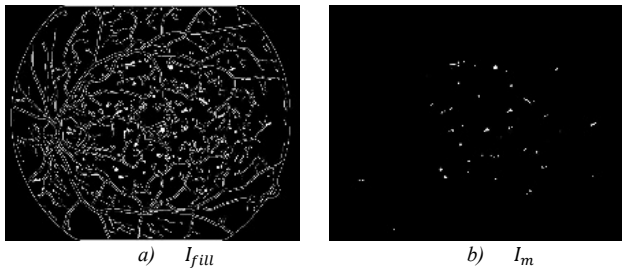


Fig. 5. Imágenes resultantes de la eliminación de vasculatura.

Dado que los MA tienen forma redondeada, se tomó en cuenta la excentricidad de los MA candidatos, los cuales varían con valores de 0 a 1, por lo que todos los MA candidatos que tuviesen mayor excentricidad a estos valores se descartaron, la imagen resultante se muestra en la Fig. 6.

Una vez obtenidos los MA candidatos se realizó la extracción de sus características, en la Tabla II se muestran los resultados promedio de las características de las 89 imágenes de fondo de ojo obtenidas de la base de datos de retinopatías diabéticas.

Para la etapa de clasificación se utilizó un algoritmo de aprendizaje supervisado KNN, el cual fue evaluado de acuerdo con sus resultados de clasificación, considerando cuatro valores: verdadero positivo (TP), falso positivo (FP), falso negativo (FN) y verdadero negativo (TN).

Los TP se obtienen cuando el clasificador detecta MA y el médico también lo detecta (acierto), los TN se obtienen cuando el clasificador no detecta MA y el médico tampoco no lo detecta (acierto), los FP se obtienen cuando el clasificador detecta MA y el médico no lo detecta (error), finalmente los FN se obtienen cuando el clasificador no detecta MA y el médico sí lo detecta (error).

TABLA II. CARACTERÍSTICAS DE LOS MA CANDIDATOS

Rango de características MA candidatos	
Características	Rango
Área	≤ 45
Perímetro	≤ 35
Excentricidad	< 1.0
Circularidad	0.5 a 2.5
Diámetro	1.5 a 4.5
Relación de aspecto	0.5 a 1

Con los valores anteriores se calcula la sensibilidad usando (8) y la especificidad mediante (9), los cuales nos indican la capacidad del sistema para discriminar los casos positivos de los negativos.

$$\text{Sensibilidad} = \frac{TP}{TP+FN} \quad (8)$$

$$\text{Especificidad} = \frac{TN}{TN+FP} \quad (9)$$

También se calcula la precisión empleando (10) y la exactitud mediante (11) la cual es el porcentaje de los MA clasificados correctamente.

$$\text{Precisión} = \frac{TP}{TP+FP} \quad (10)$$

$$\text{Exactitud} = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

En la Tabla III se muestran los resultados de la matriz de confusión obtenidos del clasificador KNN, para las 89 imágenes de fondo de ojo que contiene la base de datos de retinopatías diabéticas.

TABLA III. CLASIFICACIÓN DE MA

Matriz de confusión		
Sistema Propuesto		
Médico	Positivos	Negativos
Positivos	911	26
Negativos	165	1178

Se puede observar que 911 es el número de MA identificados con éxito y 26 es el número de MA detectados erróneamente como verdaderos por el sistema; mientras que 165 es el número de MA no detectados por el sistema y 1178 es el número de lesiones detectadas correctamente como no MA.

En la Tabla IV. se muestra una comparación de los métodos propuestos de diferentes autores y nuestro método propuesto, en la cual se puede observar una mejor sensibilidad que el método de Arati Manjaramkar et al. [5] y Rukhimini Roy et al. [6] y una mejor sensibilidad que Rukhimini Roy et al [6] y Meindert Niemeijer et al. [7].

TABLA IV. COMPARACIÓN DE LOS MÉTODOS PROPUESTOS

Autor	No de imágenes	Sensibilidad	Especificidad
Manjaramkar [5]	89	80.6%	97.5%
Roy [6]	89	89.5%	82.1%
Niemeijer [7]	89	100%	82%
Método Propuesto	89	97.2%	87.7%

A partir de estos resultados se determina que el sistema propuesto tiene una sensibilidad del 97.2%, una especificidad del 87.7%, una precisión del 84.6% y una exactitud del 91.6%.

V. CONCLUSIONES

En este trabajo se propuso un sistema para la clasificación de MA en retinografías diabéticas, logrando eliminar la vasculatura del ojo preservando los MA los cuales son de gran importancia en el diagnóstico de la RD. El sistema propuesto presentó una precisión del 84.6% y una exactitud del 91.6% evaluando el sistema con la base de datos DIARETDB1, la cual proporciona imágenes reales recopiladas por varios expertos. Como trabajo a futuro el sistema puede ampliarse para la detección de hemorragias en el fondo del ojo, así como aumentar el número de características para mejorar su sensibilidad y especificidad.

AGRADECIMIENTOS

Agradecemos al Instituto Politécnico Nacional, a la COFAA del IPN y al Consejo Nacional de Ciencia y Tecnología por el apoyo otorgado para el desarrollo de este trabajo.

REFERENCIAS

- [1] R. Klein, B. Klein, S. Moss, M. Davis, and D. DeMets, "The Wisconsin epidemiologic study of diabetic retinopathy. Prevalence and risk of diabetic retinopathy when age at diagnosis is less than 30 years," *Arch. Ophthalmol.*, vol. 102, no. 4, pp. 520–526, 1984.
- [2] Y. Ortega, Retinopatía diabética, primera causa de ceguera en país: especialistas, La jornada, Available at: "https://www.jornada.com.mx/ultimas/sociedad/2018/10/12/retinopatía-diabética-primera-causa-de-ceguera-en-país-especialistas-7717.html", [último acceso: 2020]
- [3] American Academy of Ophthalmology, ¿Que es la retinopatía diabética? [online] Available at: "https://www.aao.org/salud-ocular/enfermedades/retinopatía-diabética", [último acceso: 2020].
- [4] S. Wang, H. Lilian Tang, L. Ismail Al turk, Y. Hun, S. Sane, G. Michael Saleh and T. Peto, "Localizing Microaneurysms in Fundus Images Through Singular Spectrum Analysis", *IEEE Trans. Biomedical Engineering*, vol. 64, no. 5, pp. 990-1002, 2017.
- [5] A. Manjaramkar and M. Kokare, "A Rule Based Expert System for Microaneurysm Detection in Digital Fundus Images", 2016 International Conference on Computational Techniques in Information and

- communication Technologies (ICCTICT), New Delhi, 2016, pp 137-140.
- [6] R. Roy, S. Aruchamy and P. Bhattacharjee, "Detection of retinal microaneurysms using fractal analysis and feature extraction technique", 2013 International Conference on Communication and Signal Processing, Melmaruvathur, 2013, pp. 469-474.
- [7] M. Niemeijer, B. van Ginneken, J. Staal, M.S.A. Suttorp-Schulten and M.D. Abramoff, "Automatic Detection of Red Lesions in Digital Color Fundus Photographs", IEEE Trans. Med. Imag., vol. 24, no. 5, pp. 584-592, 2005.
- [8] Spencer, T., Olson, J. A., McHardy, K. C., Sharp, P. F., and Forrester, J. V., "An image-processing strategy for the segmentation and quantification of microaneurysms in fluorescein angiograms of the ocular fundus", Computers and biomedical research, vol. 29, no 4, pp. 284-302, 1996.
- [9] Frame, A. J., Undrill, P. E., Cree, M. J., Olson, J. A., McHardy, K. C., Sharp, P. F., and Forrester, J. V., "A comparison of computer based classification methods applied to the detection of microaneurysms in ophthalmic fluorescein angiograms", Computers in biology and medicine, vol. 28, no. 3, pp. 225-238, 1998.
- [10] Kauppi, T., Kalesnykiene, V., Kamarainen, J.-K., Lensu, L., Sorri, I., Raninen A., Voutilainen R., Uusitalo, H., Kälviäinen, H., Pietilä, J., DIARETDB1 diabetic retinopathy database and evaluation protocol, In Proc of the 11th Conf. on Medical Image Understanding and Analysis, 2007.

Sistema CAD para la Detección de Nódulos Pulmonares Basado en Características Estadísticas y de Textura

Victor Hugo Galindo R., Volodymyr Ponomaryov, Rogelio Reyes R., Clara Cruz R.

Instituto Politécnico Nacional, ESIME Unidad Culhuacán

Av. Santa Ana No. 1000, Col. San Francisco Culhuacán, CP. 04430, México, Ciudad de México

Tel. (55) 5729-6000, Ext. 73264 Fax (55) 56562058

e-mail: vgalindor1601@alumno.ipn.mx, rreyesre@ipn.mx, vponomar@ipn.mx, ccruzra@ipn.mx.

Resumen- En este trabajo se presenta un Sistema CAD (diagnóstico asistido por computadora) para la detección de nódulos pulmonares. La detección de estas masas tumorales pueden ser indicios de cáncer de pulmón, ya que los sistemas de este tipo se encargan de detectar de manera oportuna este padecimiento para contribuir en el estudio y diagnóstico por imágenes brindado por los médicos radiólogos especialistas. El sistema propuesto está dividido en cuatro etapas principales: preprocesamiento, procesamiento, extracción de características y clasificación. Las primeras dos etapas se encargan de la segmentación de las ROI (regiones de interés), y posteriormente se extraen características estadísticas y de textura de esta región. Estas características extraídas son empleadas como patrones de entrada, para una clasificación en dos posibles clases: nódulo pulmonar y no nódulo pulmonar, para lo cual se utiliza una red neuronal perceptrón. Los resultados demuestran un buen desempeño del sistema propuesto ya que obtiene un 95% de sensibilidad, 90% de especificidad y un 92.50% de precisión.

Palabras clave- Redes neuronales, nódulos pulmonares, región de interés y diagnóstico asistido por computadora

I. INTRODUCCIÓN

El término cáncer es un término técnico que se refiere al crecimiento no controlado de células en los tejidos de los órganos del cuerpo lo cual conduce a su mal funcionamiento. El estado extremo del cáncer conduce a un gran sufrimiento e incluso la muerte. De acuerdo con la Organización Mundial de la Salud (OMS) en el año 2015 se atribuyó a esta enfermedad 8,8 millones de defunciones registradas, de las cuales 571,000 fallecimientos son causados por cáncer mamario, 754,000 por cáncer gástrico, 774,000 por cáncer colorrectal, 778,000 por cáncer hepático y por último, pero no menos importante, el cáncer pulmonar con 1,690 millones de defunciones, por lo que se puede decir que es el cáncer más común que afecta tanto a hombres como a mujeres [1]. La sociedad americana contra el cáncer ha llevado un seguimiento al cáncer pulmonar, demostrando que ocurre principalmente en adultos de edad avanzada (mayores de 65 años), sin embargo, es común encontrar casos en personas de menos de 45 años que padezcan esta enfermedad mortal [2].

El estudio y análisis a través de imágenes médicas permite a los médicos especialistas observar el interior del cuerpo para encontrar indicios sobre una afección médica. Existen múltiples tipos de imágenes médicas las cuales dependen del tipo de síntoma y parte del cuerpo que deba examinarse. Para el diagnóstico de cáncer pulmonar se utilizan imágenes de tomografía computarizada (CT). Cuando se encuentra una

anormalidad en los pulmones o cerca de las áreas pulmonares significa que existe una probabilidad de que el cáncer esté presente, sin embargo, se necesitan pruebas y diagnósticos adicionales para estar completamente seguros. A las anomalías en los pulmones se les llama nódulos pulmonares, los cuales son pequeñas masas ovaladas o redondas formadas en el parénquima del pulmón llegando a medir menos de 3cm y estos pueden ser tanto benignos como malignos.

De acuerdo con los datos en “Top Doctors México” [3], el 70% de los nódulos pulmonares son benignos y el 30% son malignos. Dentro de los nódulos pulmonares benignos el 80% de las veces son granulomas, 10% malformaciones y 10% de hematomas; en el caso de los nódulos pulmonares malignos, 75% se diagnostica como carcinoma broncogénico y un 25% como metástasis, los cuales hacen referencia al cáncer.

Cuando se detectan nódulos pulmonares malignos en una etapa temprana (cuando es pequeño y no se ha propagado) es más probable que se trate médicamente con éxito, mientras que cuando no se detectan o no se le da un seguimiento, empezarán a aparecer síntomas, los cuales usualmente ocurren cuando el cáncer se encuentra en una etapa avanzada.

II. ANTECEDENTES

Dentro del campo de la detección de cáncer se tienen diferentes propuestas, las cuales desarrollan diferentes metodologías con el objetivo de realizar la detección oportuna de posibles candidatos a nódulos pulmonares.

El sistema propuesto en [4], presenta un sistema de diagnóstico asistido por computadora (CAD) basado en operaciones morfológicas para el análisis de nódulos pulmonares. El proceso consta de 4 pasos: desglose de XML, marcado de nódulos, segmentación de nódulos y una mejora de los nódulos mediante operaciones morfológicas. La etapa de preprocesamiento elimina las regiones no deseadas de la imagen CT. El sistema propuesto utiliza una interfaz gráfica, la cual permite interacción con el usuario; sin embargo, no realiza ninguna clasificación entre nódulos pulmonares

En [5] se presenta un sistema CAD, el cual trabaja de forma automática (sin ninguna intervención del usuario para su manejo), emplea un método de segmentación y detección de nódulos pulmonares basado en una red convolucional modificando la arquitectura de la red neuronal VGG-16 obteniendo una precisión del 90.55%.

El sistema CAD propuesto en [6] está constituido de 6 etapas principales para detectar nódulos pulmonares; como etapa de preprocesamiento se utiliza un filtro de mediana y posteriormente se aplica la técnica del umbral óptimo. Para extraer las ROI se utilizó la operación morfológica de apertura y clausura con un elemento de estructura de línea periódica de tamaño 2x2 muestras. Para la operación de clausura se utilizó un elemento estructural de disco de radio uno.

Los sistemas existentes cuentan en general, con las siguientes etapas: etapa de preprocesamiento, encargada de limpiar los efectos borrosos que tenga alguna imagen; etapa de segmentación, la cual permite aislar los nódulos pulmonares del fondo en las imágenes CT; etapa de extracción de características, donde se utilizan diferentes métodos para extraer distintos parámetros característicos (área, perímetro, centroide, promedio, desviación estándar, asimetría etc) y por último la última etapa de clasificación, donde se utilizan las características obtenidas como entradas a un clasificador, el cual deberá detectar la región como un nódulo pulmonar o un no nódulo pulmonar.

El sistema propuesto propone la clasificación entre nódulos pulmonares y no nódulos pulmonares por el método handcraft al igual que A. Gupta [4], con la diferencia de que la autora, solo realiza una segmentación de la ROI y no clasifica los nódulos pulmonares encontrados. Se propone el uso de la operación morfológica de apertura para la segmentación con los métodos handcraft al igual que Y. Chunran [5]; sin embargo, para el proceso de clasificación, él utiliza una red neuronal convolucional, la cual requiere demasiada memoria para lograr una segmentación de las ROI; el sistema propuesto utiliza una Red Neuronal Perceptron. Por último, J. Kaur [6], clasifica de manera paralela los dos tipos de características que se obtuvieron (características estadísticas y de textura); el sistema propuesto logra juntar ambas características.

III. SISTEMA PROPUESTO

El sistema CAD propuesto para la detección de nódulos pulmonares consta de cuatro etapas principales, las cuales se enuncian a continuación:

- 1) *Preprocesamiento*. Consta de la aplicación de un filtro de mediana y umbralización
- 2) *Procesamiento*. En esta etapa se realiza la segmentación del pulmón, aplicando la operación morfológica de apertura
- 3) *Extracción de características*. En esta etapa se extraen características de textura (energía, correlación, homogeneidad y entropía) y estadísticas (media, desviación estándar, oblicuidad y curtosis) de la ROI
- 4) *Clasificación*. Se emplea una red neuronal perceptrón para realizar una clasificación con una salida binaria (nódulo pulmonar y no nódulo pulmonar).

En la Fig.1 se muestra el diagrama a bloques del sistema propuesto.

A) Etapa de Preprocesamiento

Utilizando la base de datos LIDC (Lung Image Database Consortium) se obtiene un subconjunto de imágenes con la presencia de nódulos pulmonares de tamaño $\geq 3mm$ de tamaño

512x512 pixeles en formato DICOM (Digital Imaging and Communications in Medicine) [7]. Es un estándar internacional de imágenes médicas en escala de grises. A la imagen de entrada se le aplica un filtro de mediana, con la finalidad de reducir el ruido impulsivo presente en la imagen. Posteriormente se le aplica un umbral con valor de -500 HU, de tal forma que todos los valores que estén por debajo de este umbral serán '0' y los valores que estén por encima de este umbral serán '1'; este valor se propuso tomando en consideración la densidad que tiene un pulmón según la escala de unidades Hounsfield [8]. En la Fig. 2 se muestran algunos ejemplos de las imágenes generadas en esta etapa, img es la imagen DICOM original anteriormente mencionada de la base de datos LIDC, img_f es la imagen resultante de haber aplicado el filtro de mediana, e img_u es la imagen binaria umbralizada por HU, donde las partes blancas son estructuras de la imagen y la parte oscura son las partes no densas de la imagen.

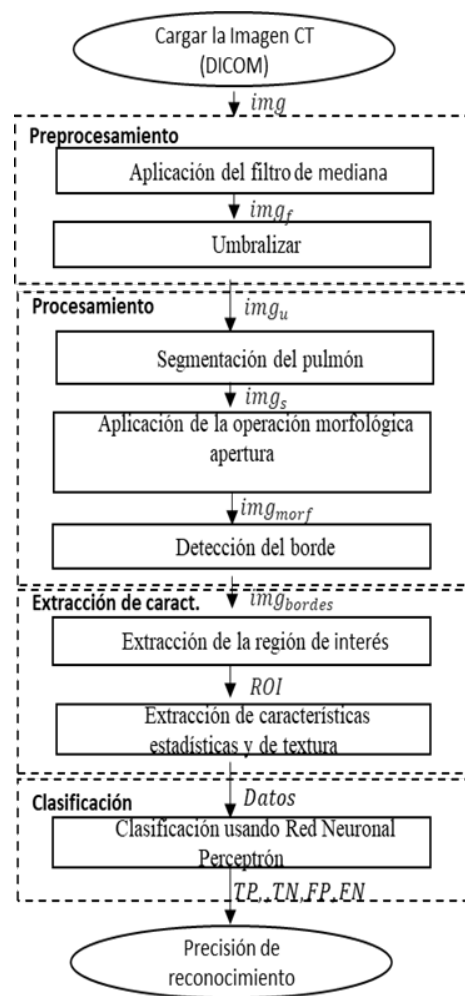


Fig. 1. Diagrama a bloques del sistema computacional.

B) Etapa de Procesamiento

Para segmentar la región del pulmón, a partir de la imagen binaria, se eliminarán aquellas áreas superiores a 490 pixeles blancos, tomando bloques de 23x23 pixeles de la imagen. Con el objetivo de tener una imagen totalmente segmentada, para lo

cual se utilizará la operación morfológica de apertura, esta operación requiere un elemento estructural (kernel), el cual tiene una forma de disco con un radio de tamaño 6. Dada una imagen binaria A y un elemento estructural binario K la apertura se define como (1), donde se aplica una erosión entre A y K y posteriormente una dilatación con el elemento estructural K .



Fig. 2. Imágenes resultantes de la etapa de preprocesamiento. a) Imagen original DICOM 512x512 píxeles; b) Imagen filtrada; y c) Imagen binaria umbralizada por HU.

$$AoK = (A \ominus K) \oplus K \quad (1)$$

Esta operación se encarga de suavizar los contornos de una imagen. También puede eliminar franjas o zonas de un objeto que sean “más estrechas” que el elemento estructural [10].

Posteriormente a la eliminación de áreas, se aplica el operador de Sobel para obtener los bordes de la imagen, la cual será de ayuda para determinar si un nódulo es benigno (bordes regulares) o maligno (bordes irregulares). El operador Sobel utiliza una máscara de 3x3 elementos, que se mueve píxel a píxel, calculando el valor del gradiente ∇f utilizando las segundas derivadas en ambas direcciones $\frac{\partial f^2}{\partial x}$ y $\frac{\partial f^2}{\partial y}$ como en (2).

$$\nabla f = \sqrt{\left[\frac{\partial f^2}{\partial x} + \frac{\partial f^2}{\partial y} \right]}. \quad (2)$$

En la Fig. 3 se muestra el resultado de haber segmentado la región del pulmón de la imagen img_s , la imagen resultante de la operación morfológica con el kernel de tipo disco y tamaño de 6 píxeles es img_{morf} y finalmente la imagen obtenida utilizando el operador de Sobel es img_{bordes} .

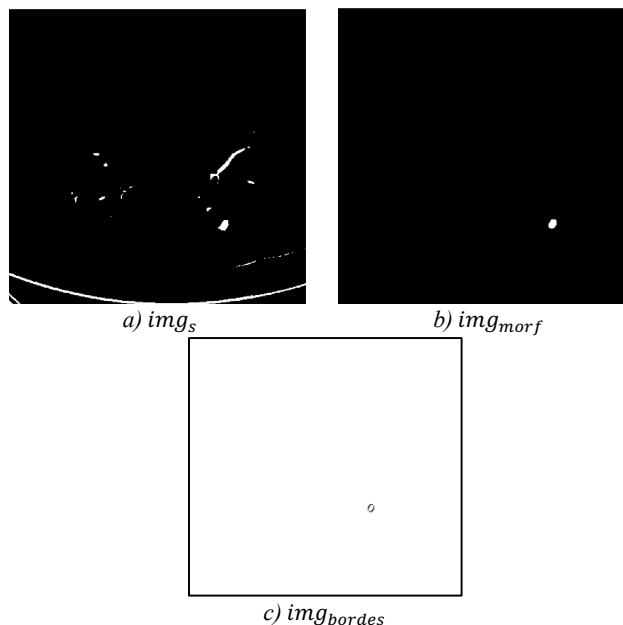


Fig. 3. Imágenes resultantes de la etapa de procesamiento. a) Imagen segmentada; b) Imagen morfológica; y c) Imagen con la detección de bordes.

C) Etapa de Extracción de Características

A partir de la imagen img_{morf} se extrae la región de interés (ROI) y posteriormente se realiza el cálculo del área y de sus características estadísticas y de textura. Para extraer la región de interés se requiere cortar la imagen a través de sus bordes (este tamaño puede variar dependiendo del tamaño de la masa pulmonar encontrada), como se muestra en la Fig. 4.



Fig. 4. Extracción de la ROI.

Utilizando este tipo de imágenes generadas con la región de interés totalmente extraída, se procede a la extracción de características de textura como la energía, correlación, homogeneidad y entropía, y también características estadísticas tales como la media, desviación estándar, oblicuidad y curtosis; finalmente estos valores característicos obtenidos de la ROI se emplean para generar vectores de características, los cuales serán la entrada al clasificador.

D) Etapa de Clasificación

El objetivo de la última etapa es clasificar las características en dos diferentes etiquetas. Siendo la primera la presencia de algún nódulo pulmonar y la segunda de un no nódulo pulmonar. Para tal efecto, se emplea una red neuronal perceptrón, donde se

utiliza el 80% de los vectores de características generadas (también llamados vectores característicos), para entrenar a la red neuronal y el 20% restante para realizar pruebas de verificación; una vez hecho esto, se procede a evaluar la precisión de la red neuronal. De manera general el algoritmo de entrenamiento de la red neuronal empieza generando una matriz de pesos aleatoria W con un tamaño de 9 neuronas de entrada, debido a que es el tamaño de los patrones característicos de entrada llamados $X(t)$. Se procede a calcular la salida $Y(t)$ mediante (3) y la función de transferencia (4).

$$Y(t) = ft(W * X(t)). \quad (3)$$

$$ft = \begin{cases} 1 & \text{si } Y(t) \geq 0 \\ -1 & \text{si } Y(t) < 0 \end{cases} \quad (4)$$

Posteriormente se calcula el error entre la salida esperada $S(t)$ y la salida real $Y(t)$ mediante (5), y este valor es usado para adaptar los nuevos pesos de la red neuronal empleando (6)

$$e(t) = S(t) - Y(t), \quad (5)$$

$$W(t + 1) = W(t) + [\alpha * e(t) * X(t)]. \quad (6)$$

En la Fig. 5 se puede observar la arquitectura empleada donde la capa de entrada recibe tantas características como el tamaño de los patrones característicos. La capa de salida corresponde a la clasificación de la red neuronal además de contar con el algoritmo de adaptación.

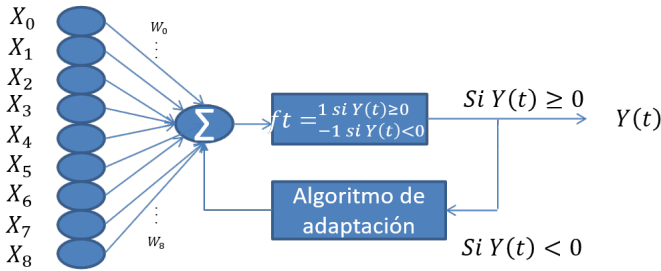


Fig. 5. Arquitectura de la red neuronal perceptrón

IV. RESULTADOS

El sistema CAD propuesto fue probado empleando imágenes de tomografía computarizada obtenidas de la base de datos LIDC en formato DICOM que presentan nódulos pulmonares e imágenes que no los presentan. La Fig. 6 muestra ejemplos de imágenes binarizadas resultantes de la etapa de preprocesamiento, en donde se observa la delimitación correcta de la región de los pulmones.

En la Fig. 7 se pueden observar 4 ejemplos de imágenes morfológicas, resultantes de la etapa de preprocesamiento correspondientes a las 4 imágenes de la Fig. 6, en las cuales se observan los nódulo aislados para determinar si éste, es benigno o maligno.

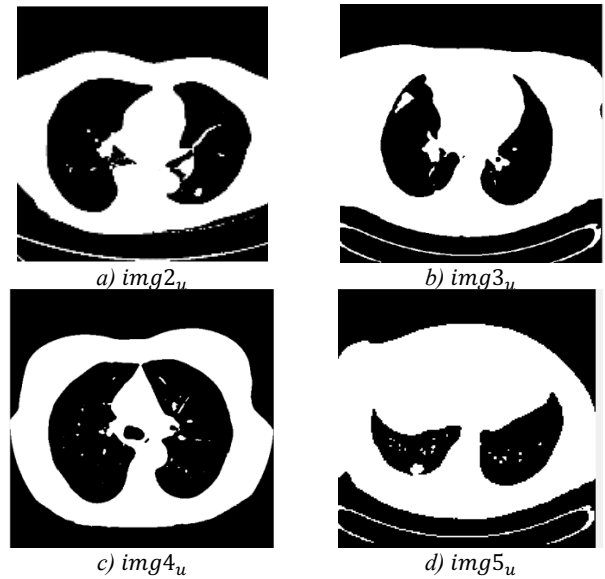


Fig. 6. Imágenes binarizadas de diferentes imágenes DICOM, generadas en el preprocesamiento.

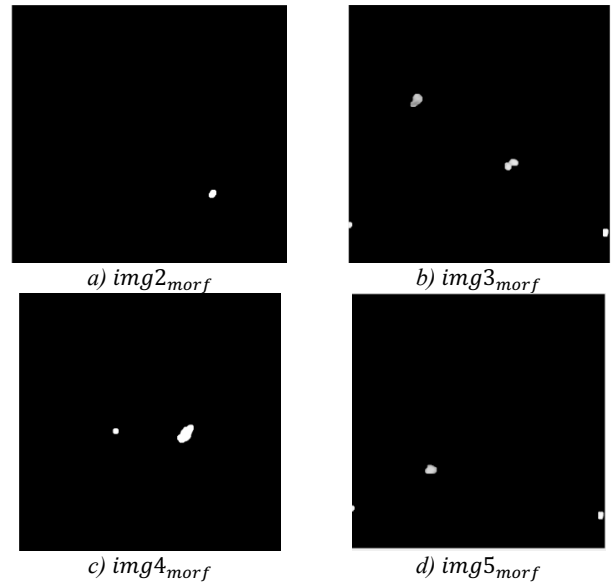


Fig. 7. Imágenes morfológicas generadas en la etapa de procesamiento, correspondientes a las 4 imágenes de la Fig. 6.

La Fig. 8 muestra las regiones de interés extraídas de la etapa de extracción de características pertenecientes a las cuatro imágenes de la Fig. 7; estas imágenes corresponden a posibles nódulos pulmonares, a los cuales se les extraen sus características para realizar su clasificación. El tiempo promedio de ejecución de las etapas de preprocesamiento, procesamiento y extracción de características para cada imagen es de 5.92 segundos.

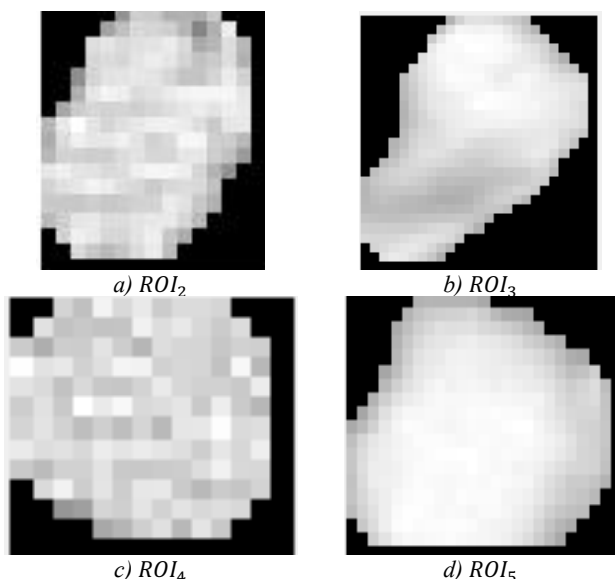


Fig. 8. ROI generadas en la etapa de extracción de características, correspondientes a las 4 imágenes de la Fig. 7.

Los criterios de evaluación empleados para determinar el grado de generalización y correcto funcionamiento del sistema son la sensibilidad, la especificidad y la precisión; estos cálculos se realizan de acuerdo con las diferentes relaciones entre los TP (Verdaderos Positivos), FN (Falsos Negativos), FP (Falsos Positivos) y TN (Verdaderos Negativos).

Supongamos un evento E que tendrá dos posibles resultados los cuales tomarán los valores de la siguiente forma:

$$E = \begin{cases} 1 & \text{si la persona tiene cancer} \\ 0 & \text{si la persona no tiene cancer} \end{cases}$$

TP son aquellos casos cuando los datos reales son 1 (verdadero) y la predicción también tiene un valor de 1 (verdadero). TN son aquellos casos cuando los datos reales son 0 (falso) y la predicción también tiene un valor de 0 (falso). FP son aquellos casos en los que los datos reales indican que es 0 (falso) y la predicción indica que es 1 (verdadero), es decir la predicción ha sido errónea. FN son aquellos casos en los que los datos reales indican que es 1 (verdadero) y la predicción indica que es 0 (falso) ocasionando que la predicción ha sido errónea.

Considerando los cuatro posibles casos TP , FN , FP y TN se pueden calcular los siguientes criterios de evaluación.

1) *Sensibilidad*: corresponde a la proporción de hechos correctamente clasificados con la enfermedad realizados por un diagnóstico. Cuanto mayor es la sensibilidad de la prueba, más enfermos serán diagnosticados adecuadamente y la tasa de FN será menor, se calcula empleando (7)

$$\text{Sensibilidad} = \frac{TP}{TP+FN} \quad (7)$$

2) *Especificidad*: corresponde a la proporción de hechos reconocidos como negativos. Cuanto mayor es la especificidad de la prueba, más sanos serán diagnosticados adecuadamente, con lo que la tasa de FP será menor y se calcula empleando (8)

$$\text{Especificidad} = \frac{TN}{TN+FP} \quad (8)$$

3) *Precisión (accuracy)*: mide el porcentaje de casos que el modelo clasifica de manera correcta y se calcula usando (9)

$$\text{Precisión} = \frac{TP+TN}{TP+FP+TN+FN} \quad (9)$$

La Tabla I muestra los valores promedio de sensibilidad, especificidad y precisión obtenidos al evaluar el sistema propuesto, donde el valor de sensibilidad es del 95%; esto representa una alta predicción del sistema cuando la imagen DICOM corresponde a un paciente con cáncer. La especificidad del 90%, determina los casos cuando la imagen corresponde a un paciente que no tiene cáncer y finalmente, la precisión determina los casos que el sistema propuesto clasifica de manera correcta obteniendo un porcentaje del 92.50%,

TABLA I. CRITERIOS DE EVALUACIÓN DEL SISTEMA

Sensibilidad	Especificidad	Precisión
95%	90%	92.50%

En la Tabla II se muestran una comparativa con los resultados de trabajos del estado del arte; los criterios evaluados son sensibilidad, especificidad y precisión. El sistema propuesto propone la clasificación entre nódulos pulmonares y no nódulos pulmonares por el método handcraft y utiliza la operación morfológica de apertura para la segmentación con los métodos handcraft al igual que Y. Chunran [5]; sin embargo, para el proceso de clasificación, él utiliza una red neuronal convolucional, la cual requiere demasiada memoria para lograr una segmentación de las ROI; mientras que, el sistema propuesto utiliza una Red Neuronal Perceptron. J. Kaur [6], clasifica de manera paralela los dos tipos de características que se obtuvieron (características estadísticas y de textura), el sistema propuesto logra conjuntar ambas características para ser clasificadas y obtener una precisión del 92.5%

TABLA II. COMPARACIÓN CON EL ESTADO DEL ARTE

Autor	Sensibilidad	Especificidad	Precisión
Y. Chunran [5]	-	-	90.55%
J. Kaur (Estadísticos) [6]	90%	99%	94.5
J. Kaur (Textura) [6]	99%	98%	98%
Sistema Propuesto	95%	90%	92.5%

V. CONCLUSIONES

La detección de masas tumorales pueden ser indicios de cáncer de pulmón, por lo que la detección automática y oportuna de estos, contribuye a los médicos radiólogos especialistas a dar un tratamiento oportuno para la erradicación de la enfermedad. Los nódulos pulmonares pueden ser detectados y clasificados de dos tipos: como nódulo pulmonar y como una estructura del cuerpo parecida a un nódulo pulmonar, como pueden ser granulomas, malformaciones, hematomas, etc. El sistema CAD propuesto es capaz de detectar nódulos pulmonares mediante una clasificación binaria, además de obtener y conservar la información original de las regiones de interés (ROI). Los resultados obtenidos demuestran que, mediante los algoritmos y las técnicas de procesamiento de imágenes utilizadas, se

obtienen buenos resultados en cada una de las etapas del sistema computacional para la detección de nódulos pulmonares. El sistema propuesto obtiene un porcentaje de clasificación del 92.5% gracias a los algoritmos handcraft utilizados. El sistema propuesto realiza una nueva clasificación entre los nódulos pulmonares y los no nódulos pulmonares conjuntando ambos tipos de características que se tienen de estas masas tumorales.

AGRADECIMIENTOS

Agradecemos al Instituto Politécnico Nacional, a la COFAA del IPN, al Programa Institucional de Formación de Investigadores del IPN y al Consejo Nacional de Ciencia y Tecnología por el apoyo otorgado para el desarrollo de este trabajo.

REFERENCIAS

- [1] O. M. d. I. Salud, "Who," 12 Septiembre 2018. [En línea]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/cancer>. [Último acceso: 2021].
- [2] A. C. Society, "Cancer.org," 14 Enero 2020. [En línea]. Available: <https://www.cancer.org/es/cancer/cancer-de-pulmon/acerca/estadisticas-clave.html>. [Último acceso: 2021].
- [3] D. F. B. Morales, "Top Doctors," 17 12 2015. [En línea]. Available: <https://www.topdoctors.mx/articulos-medicos/nodulo-pulmonar-lesion-asintomatica-y-benigna>. [Último acceso: 2021].
- [4] A. Gupta, O. Märtens, Y. Le Moullec and T. Saar, "A tool for lung nodules analysis based on segmentation and morphological operation," 2015 IEEE 9th International Symposium on Intelligent Signal Processing (WISP) Proceedings, Siena, 2015, pp. 1-5.
- [5] Y. Chunran, W. Yuanvuan and G. Yi, "Automatic Detection and Segmentation of Lung Nodule on CT Images," 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Beijing, China, 2018, pp. 1-6.
- [6] J. Kaur, N. Garg and D. Kaur, "An automatic CAD system for early detection of lung tumor using back propagation network," 2014 International Conference on Medical Imaging, m-Health and Emerging Communication Systems (MedCom), Greater Noida, 2014, pp. 257-261.
- [7] T. M. I. T. A. (MITA), «dicomstandard.» 2020. [En línea]. Available: <https://www.dicomstandard.org/about-home>. [Último acceso: 2021].
- [8] S. E. d. I. Cardiaca, "ecocardio.com," [En línea]. Available: <https://ecocardio.com/documentos/biblioteca-preguntas-basicas/preguntas-al-radiologo/892-que-son-coeficiente-atenuacion-numero-hounsfield-y-ventana.html>. [Último acceso: 2021].
- [9] Richard. E. Rafael C. Gonzalez, Digital Image Processing 2da ed, New Jersey, Prentice Hall, 2002.
- [10] RadiologyInfo, "RadiologyInfo," 20 Enero 2018. [En línea]. Available: <https://www.radiologyinfo.org/sp/info.cfm?pg=chestrad>. [Último acceso: 2021].
- [11] C. Cloud, "Clinic Cloud," [En línea]. Available: <https://clinic-cloud.com/blog/formato-dicom-que-es-estandar-imagenes-medicas/>. [Último acceso: 2021].

Detección e identificación de manchas en paneles solares con técnicas de visión artificial

Flores Herrera Viviana
Tecnológico Nacional de México-
Instituto Tecnológico de La Paz
La Paz Baja California Sur,
México
viviana7robot@gmail.com

Martínez Díaz Saúl
Tecnológico Nacional de México-
Instituto Tecnológico de La Paz
La Paz Baja California Sur,
México
saul.md@lapaz.tecnm.mx

Gutiérrez Jagüey Joaquín
Centro de Investigaciones
Biológicas del Noroeste
La Paz Baja California Sur,
México
joaquina04@cibnor.mx

Santillán Méndez Israel Marcos
Tecnológico Nacional de México-
Instituto Tecnológico de La Paz
La Paz Baja California Sur,
México
israel.sm@lapaz.tecnm.mx

Resumen— En este artículo se presenta el desarrollo de un sistema de visión artificial para detectar e identificar manchas en módulos solares. El sistema se divide en dos partes. Una clasificación del panel solar que determina si el panel solar está limpio o sucio y la detección e identificación de manchas en la superficie de este. Se tomaron fotografías de paneles limpios y sucios a lo largo de 50 semanas. La clasificación se realizó con máquinas de soporte vectorial para determinar si el panel está limpio o sucio. Se generó una base de datos de manchas capturadas con una cámara ESP-32 CAM a una altura de 15 cm y a partir de estas imágenes se utilizó OpenCV para filtrar y procesar para su identificación. Para probar la eficiencia de los procedimientos propuestos se eligieron 50 imágenes de los paneles solares al azar. El sistema identificó las manchas en 40 imágenes (80 %). Los algoritmos descritos en este trabajo resultan efectivos para la tarea de clasificación de paneles limpios o sucios, así como para identificar manchas en paneles solares con el fin de aplicar estrategias de limpieza. Se planea que en un futuro estos algoritmos sean implementados para controlar un robot autónomo limpiador de paneles solares.

Palabras clave—paneles solares, visión artificial, identificación de manchas

I. INTRODUCCIÓN

Debido a reservas de combustibles fósiles finitas, un potencial hidroeléctrico limitado y una creciente demanda de electricidad, como resultado del crecimiento de la población mundial, se predice que a mediano y principalmente a largo plazo la electricidad tendrá que generarse a partir de nuevas fuentes para mantener un equilibrio entre la oferta y la demanda, tanto a nivel nacional como global [1]. Por esta razón, se ha optado por la utilización de tecnologías limpias para la generación de energía eléctrica con el aprovechamiento de fuentes renovables de energía y la conservación del medio ambiente [2].

En el contexto de energía renovable, la energía solar se ha convertido en una de las opciones más dominantes y preferidas, ya que los paneles solares pueden ser instalados en regiones donde la red eléctrica no tiene alcance y son tanto versátiles como flexibles para ser adaptadas a diferentes requerimientos

[1]. Las celdas solares son las encargadas de recibir la radiación solar y generar electricidad. En términos de funcionamiento, los paneles solares deben exponerse a la intemperie; en consecuencia, polvo, excrementos de aves, aerosoles marinos, el polen y, por supuesto, la contaminación son fenómenos externos que puede depositarse gradualmente en la superficie de los paneles solares [3].

Un panel solar comercial tiene una eficiencia del 15 %, aproximadamente una sexta parte de la luz solar que incide en el área del panel [2]. El polvo acumulado en su superficie puede reducir la eficiencia del sistema hasta en un 50 %, por esta razón los paneles solares requieren mantenimiento después de su instalación [1]. Cuando un panel solar se encuentra sucio se presentan tres posibles efectos contraproducentes: reducción de la potencia de salida, disminución de la eficiencia general y formación de puntos calientes y celdas dañadas [4]. Cuando algunas de las celdas tienen manchas, el área sombreada no genera suficiente energía para igualar las otras celdas y actúan como carga muerta. La alta temperatura del área sombreada por las manchas aumenta la formación de puntos calientes y el módulo solar puede sufrir daños permanentes [3]. Por estas razones se recomienda limpiar los paneles solares de manera preventiva y rutinaria. Esta tarea puede volverse tediosa o cansada para quienes la realizan ya que, por lo general, no se instala solo un panel solar sino un conjunto de varios paneles. Además, los paneles solares no solo pueden tener polvo, sino también manchas o suciedad profunda que puede requerir más esfuerzo a la hora de realizar la tarea de limpieza.

Para reducir el tiempo y el esfuerzo, la limpieza de paneles se puede hacer mediante sistemas automáticos o autónomos, clasificados en tres tipos: sistemas de rieles, sistemas de robot atado y sistemas de robot móviles. En la primera clase se instalan rieles a los paneles solares y un cepillo que se desplaza por ellos rieles, la idea es pasar el cepillo a una cierta velocidad para eliminar el polvo sin considerar las manchas. Tal es el caso del sistema Ecoppia E4 [4] que no utiliza agua y cuenta con un cepillo con flujo de aire. Este tipo de sistemas está unido de forma permanente al panel solar. Los sistemas atados emplean cables con el fin de mover y controlar al robot. Desde la parte

superior del panel, para un mejor funcionamiento estos sistemas necesitan trabajar en paneles con inclinación [5]. Por último, los sistemas de robot móviles no requieren instalación en los paneles solares, ya que, por lo general, estos robots utilizan diversos sensores y algoritmos con rutas establecidas para llevar a cabo la tarea; su principal objetivo es realizar y finalizar el recorrido por el panel solar sin considerar la eficiencia en la limpieza. Están programados para limpiar siempre a una velocidad constante, sin tener en cuenta las condiciones del área que se desea limpiar [6].

Por otra parte, la visión artificial permite adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información [7]. Un campo donde la visión artificial tiene gran participación es la robótica y los vehículos autónomos. En estas áreas se utiliza para localizar obstáculos, encontrar el camino e identificar objetos o personas, entre otras aplicaciones. Para lograr su objetivo los robots emplean cámaras y sensores activos para obtener una imagen de su entorno, una vez que es obtenida la imagen se interpreta y se extrae la información.

En este trabajo se propone el desarrollo de un sistema de visión artificial para realizar dos tareas: primero clasificar si un panel solar está limpio o sucio de polvo, y como segunda actividad es detectar e identificar manchas para controlar en un futuro un robot autónomo limpiador de módulos solares. La idea principal es que el robot además de remover el polvo pueda identificar las manchas en el módulo solar y estimar el tiempo que le tomaría quitar la mancha con su herramienta de limpieza, considerando para esto su nivel de energía y así decidir cuánto tiempo invertir en quitar la mancha o si es factible realizar el recorrido por el panel solar retirando el polvo y dedicarle un menor tiempo a la mancha. Se espera que el consumo de energía del sistema de visión artificial en conjunto con la plataforma móvil propuestos sea menor a la energía que se obtiene al mantener los paneles limpios con respecto sucios. Sin embargo, en este trabajo no se presentan resultados del consumo de energía del sistema.

II. DESCRIPCIÓN DEL SISTEMA

La Figura 1 muestra las unidades que conforman el sistema de visión artificial. Una cámara digital, la cual está montada en una plataforma móvil, basada en la tarjeta ESP-32 CAM para generar el banco de imágenes. Así como los algoritmos para clasificar y detección.

A. Plataforma móvil

Para realizar este trabajo se cuenta con una plataforma móvil la cual tiene dos ruedas tipo oruga. Esta plataforma es capaz de realizar un recorrido por el panel solar con una ruta preestablecida. A la plataforma base se le instaló una cámara

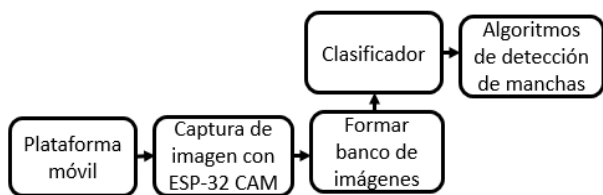


Figura 1. Diagrama general del sistema.

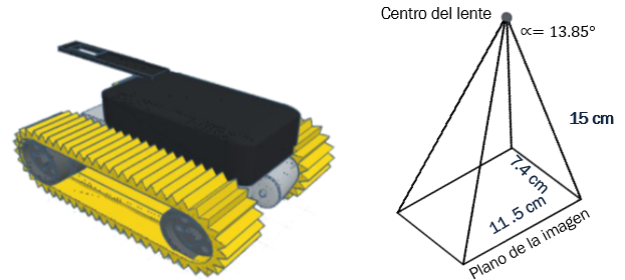


Figura 2. Plataforma móvil.

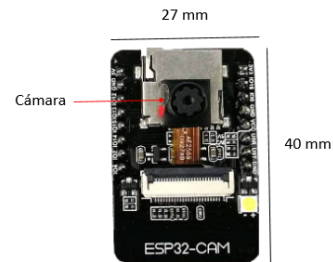


Figura 3. Tarjeta electrónica ESP-32 CAM.

incorporada en la tarjeta ESP-32 CAM. La cámara fue instalada a una altura de 15 cm paralela a la superficie de panel solar para obtener imágenes de 7.4 cm por 11.5 cm [Fig. 2].

B. Tarjeta ESP-32 CAM

Para tomar las imágenes desde la plataforma se utilizó la tarjeta programable ESP-32 CAM [Fig. 3], la cual cuenta con una cámara modelo OV2640 de 2 megapíxeles con flash incluido y con un ángulo de visión α de 13.85°. Esta tarjeta permite guardar la imagen capturada por la cámara en una memoria SD o transmitir la imagen hacia otros dispositivos o tarjetas electrónicas mediante Wi-Fi. Se puede programar en diversos lenguajes, para este trabajo se programó en C++.

C. Banco de imágenes

El banco de imágenes consta de dos diferentes grupos, uno para trabajar en la clasificación de paneles limpios y sucios de polvo y otro para la detección de manchas. Estas imágenes sirven para evaluar los algoritmos, aunque el objetivo al final es que este sistema de visión opere en tiempo real sobre la plataforma base descrita anteriormente. En el primer grupo de imágenes fue de 80 muestras del mismo modelo de panel solar: 40 de paneles limpios y 40 de paneles sucios en diferentes condiciones de iluminación. Para obtener las muestras, se recortaron las fotos de paneles solares como las que se aprecian



Figura 4. Imagen del panel solar con sección limpia y sección sucia por polvo.

en la figura 4, donde se observan partes limpias de los paneles solares y partes sucias.

Las imágenes de muestras son de 300 x 300 píxeles, la idea es hacer un clasificador utilizando máquinas de soporte vectorial (MVS) para identificar paneles limpios y sucios. En la figura 5 se presenta un ejemplo de las imágenes utilizadas como entrada al sistema, los paneles de la izquierda están limpios y los de la derecha sucios.

El segundo banco de imágenes contiene diferentes tipos de manchas [Fig. 6], las cuales se obtuvieron desde la plataforma base donde se montó la tarjeta ESP-32 [Fig. 2]. Actualmente se cuenta con aproximadamente 500 imágenes obtenidas del mismo panel solar en diferentes condiciones de iluminación. Se realizaron diferentes pruebas con el objetivo de encontrar las condiciones óptimas de iluminación que permitan la mayor identificación de manchas en el panel solar. Se tienen diferentes tipos de imágenes: imágenes con manchas [Fig. 7(a)], sin manchas, [Fig. 7(b)] tomadas en diferentes horarios entre 7 y 8 a.m. [Fig. 7(e)] [Fig. 7(f)] y entre 7 y 8 p.m. [Fig. 7(c)] [Fig. 7(d)], imágenes con flash [Fig. 7(g)] e imágenes sin flash [Fig. 7(h)].

III. CLASIFICADOR DE PANELES LIMPIOS O SUCIOS

Para detectar si un panel solar está limpio o está sucio se utilizó una MVS. Este modelo de clasificación supervisada permite clasificar los datos buscando un hiperplano con mayor margen entre dos clases.

A. Características

Para obtener las características se trabajó con el primer banco de imágenes; se utilizó el histograma de cada componente RGB de las imágenes, así como el histograma en grises. Al comparar los histogramas de un panel limpio y un panel sucio, se puede apreciar como hay una diferencia en el punto donde se presenta el pico de la curva y utilizarlo como característica. Se empleó, entonces, el valor de x donde se encuentra el máximo



Figura 5. Imágenes de muestras de 300 x 300 píxeles.

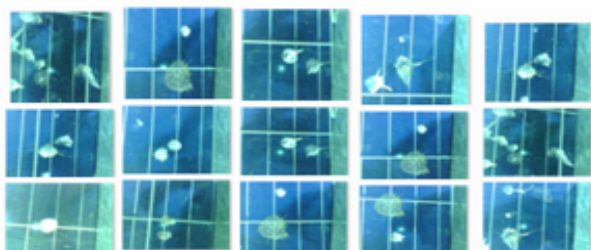


Figura 6. Banco de imágenes.

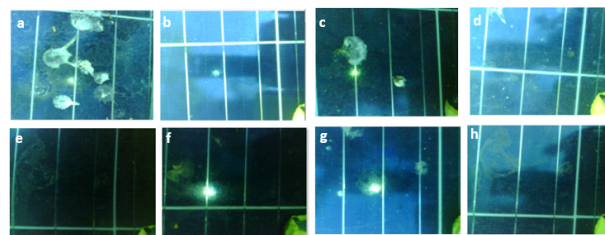


Figura 7. Tipos de imágenes: a) con manchas, b) sin manchas, c) capturadas a las 8 pm, d) capturadas las 7 pm, e) capturadas a las 8 am, f) capturadas a las 7 am, g) con flash, y h) sin flash.

de cada componente, es decir, el valor del pixel que más se repite (0-255, el punto máximo del histograma). Se probó utilizar los valores en y , que representa la frecuencia con que se presenta cada pixel, pero no aportaba información útil como característica [Fig. 8].

B. Clasificador basado en Máquinas de soporte vectorial

Se implementó un clasificador basado en Máquina de Soporte Vectorial implementado en Matlab para identificar si un panel está limpio o sucio. Para este trabajo se utilizó el 70% de los datos para entrenar y el 30% para pruebas. Se probaron dos configuraciones de kernel, una lineal donde se aplicó la ecuación (1) y otra de base radial de la ecuación (2). Donde x_1 y x_2 son los vectores de características.

$$K(x_1, x_2) = x_1^T x_2 \quad (1)$$

$$K(x_1, x_2) = \exp\left(\frac{\|x_1 - x_2\|^2}{2\sigma^2}\right) \quad (2)$$

IV. DETECCIÓN DE MANCHAS

Se utilizó Python 3 y OpenCV 4 para implementar un algoritmo propio de identificación de manchas, utilizando el segundo banco de imágenes descrito anteriormente.

La figura 9 muestra el diagrama del algoritmo propuesto para identificar las manchas en los paneles solares utilizando el procesamiento de imágenes. Se considera que existen diferentes tipos de manchas dependiendo de las condiciones de iluminación, ya que los paneles solares están instalados al aire libre. Por ello, se tienen dos estrategias para identificar manchas,

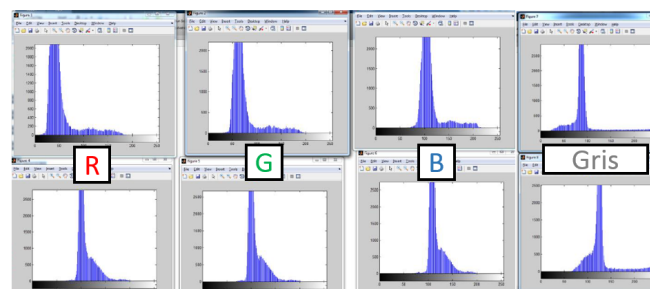


Figura 8. Características de los histogramas, de una imagen con un panel limpio (fila superior) y un panel sucio (fila inferior).

básicamente se trata del mismo proceso, pero cambia en sensibilidad de detección. Uno de ellos se aplica sobre la imagen de entrada convertida a grises y la otra se procesa como si fuera una imagen en grises, pero utilizando únicamente el componente rojo de la imagen de entrada. A la imagen RGB original se le realiza un pre procesamiento para eliminar la cuadrícula que presentan todos los paneles solares en su superficie. Después se aplica el procesamiento principal que toma como base la imagen de entrada convertida a grises. Si no se identifican manchas después de este procesamiento, pasa por el segundo procesamiento donde se considera únicamente el componente rojo de la imagen original.

A. Pre procesamiento

Se realizaron algunas pruebas en los diferentes canales de color de la imagen y en escala de grises [Fig. 10] donde se observó que la imagen en el canal rojo y gris tenía menos afectaciones de iluminación respecto a las imágenes en el canal verde y azul.

El canal rojo en particular, presenta el mayor contraste entre las líneas de la cuadrícula del panel solar y el fondo oscuro, por lo que se eligió para realizar el pre procesamiento.

En la figura 11 se muestra el diagrama del pre procesamiento, este permite preparar la imagen para detectar las manchas.

A la imagen en grises que se obtiene de considerar solo el componente rojo, se aplica un filtro para reducir el ruido. De las ventajas de aplicar un filtro Gaussiano con respecto de un filtro de caja es que el kernel da más peso al centro y menos a los bordes [Fig.12]. Como parámetros de entrada se establecieron el ancho y la altura del kernel que es un número positivo e impar. También se define la desviación estándar en las



Figura 11. Pre procesamiento de imágenes.

$$\begin{bmatrix} 0 & 1 & 2 & 1 & 0 \\ 1 & 3 & 5 & 3 & 1 \\ 2 & 5 & 9 & 5 & 2 \\ 1 & 3 & 5 & 3 & 1 \\ 0 & 1 & 2 & 1 & 0 \end{bmatrix}$$

Figura 12. Matriz de filtro Gaussiano de 5x5.

direcciones X y Y, sigma X y sigma Y. En este trabajo la desviación estándar se calculó a partir del tamaño del kernel.

Para binarizar la imagen filtrada se utiliza la umbralización, una técnica de segmentación que permite separar los pixeles de una imagen en escala de grises en dos categorías a partir de un valor umbral de intensidad [8]. En este trabajo para convertir la imagen de grises a binaria se utilizó la umbralización de Otsu. Este método determina un valor umbral global óptimo a partir del histograma de la imagen. Para ello toma un valor aproximado entre los dos picos del histograma.

A fin de realizar la detección de bordes se emplea el algoritmo de Canny el cual se divide en 3 etapas. La primera etapa utiliza la detección de bordes con Sobel, para esto se emplea un kernel Sobel en la dirección horizontal (I_x) y otro para la dirección vertical (I_y), donde la suma de los pesos es igual a cero, que corresponde con la derivada de una función constante [Fig. 13]. El resultado de aplicar estos kernel son los bordes de la imagen y la dirección del gradiente de la imagen original. La magnitud del gradiente aproximado está dada por (3) y donde (4) indica la dirección perpendicular de los bordes [8].

$$\sqrt{I_x(x, y)^2 + I_y(x, y)^2} \approx \|\nabla I(x, y)\|_2 \tag{3}$$

$$\theta = \tan^{-1}\left(\frac{I_y}{I_x}\right) \tag{4}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} / 2 \quad \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} / 2$$

Figura 13. Kernel Sobel I_x y I_y .

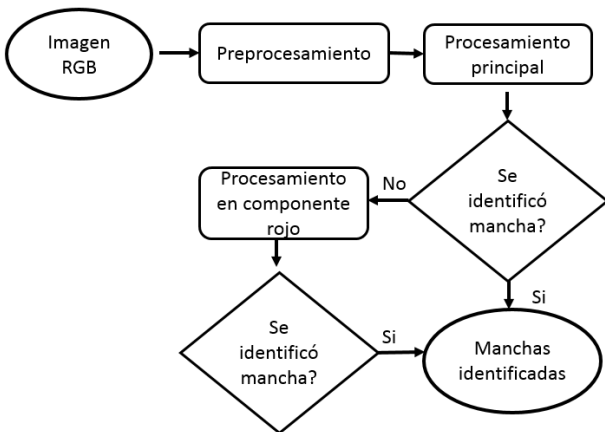


Figura 9. Diagrama general del sistema de visión artificial

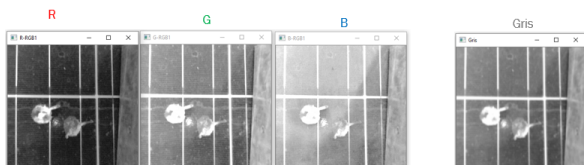


Figura 10. Imagen en los canales RGB y escala de grises.

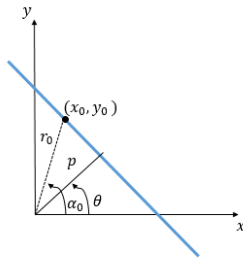


Figura 14. Punto (x_0, y_0) en coordenadas polares.

En la segunda etapa se hace una eliminación de píxeles fuera de borde. Esta etapa permite contar solo con los píxeles que cumplen la condición de tener un grosor de 1. Por último, se aplica un umbral de histéresis, que se caracteriza por establecer dos umbrales, uno máximo y uno mínimo. Si el valor del píxel es mayor que el umbral máximo es considerado parte del borde. Si el valor del píxel es menor que el umbral mínimo se considera que no forma parte del borde. Si el valor del píxel está entre el valor máximo y el mínimo se considera parte del borde solo si está conectado a otro píxel que ya sea parte del borde.

En la detección y eliminación de líneas lo que se busca es eliminar de las imágenes las líneas rectas que tienen todos los paneles solares en forma de rejilla y así poder identificar las manchas y no las líneas. Para identificarlas, se utiliza la transformada de Hough, la cual hace agrupaciones de los puntos que pertenecen a los bordes de posibles líneas a través de un procedimiento de votación sobre un conjunto de líneas parametrizadas. Para poder representar todas las posibles rectas que hay en la imagen se utiliza la ecuación de la recta en coordenadas polares (5).

$$p = x \cos \theta + y \sin \theta \quad (5)$$

Para un punto arbitrario en la imagen con coordenadas (x_0, y_0) , las rectas que pasan por ese punto son los pares (p, θ) donde p es la distancia entre la línea y el origen y θ el ángulo de la normal desde el origen, r_0 es la distancia del punto (x, y) al origen. Puede variar entre los valores $[0, \infty)$ y α_0 es el ángulo que forma el vector r con el eje vertical de las x . Esto corresponde a una curva sinusoidal en el espacio (p, θ) que es única para ese punto [9] [Fig. 14].

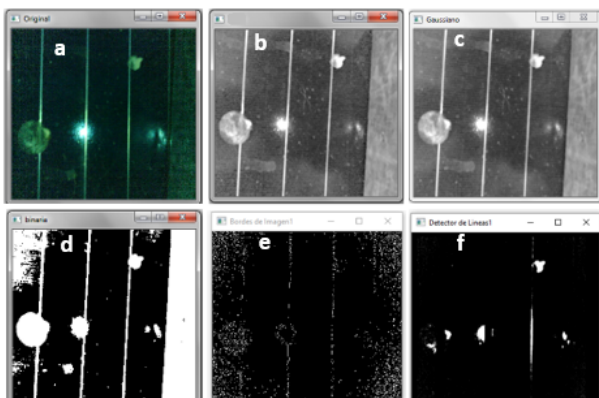


Figura 15. Resultados del preprocesamiento: a) Imagen original, b) componente rojo, c) filtro Gaussiano, d) umbralización de Otsu, e) detección de bordes, y f) detección y eliminación de líneas.

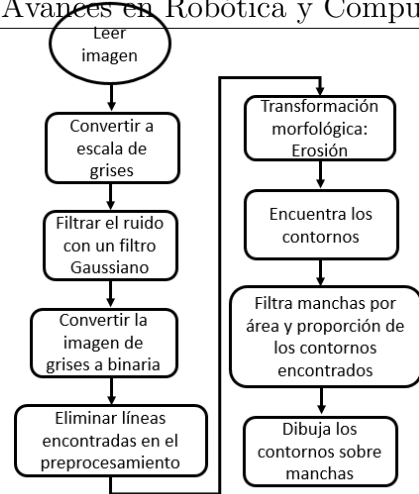


Figura 16. Algoritmo de Procesamiento Principal.

Si las curvas correspondientes a dos puntos se intersectan, el punto de intersección en el espacio de Hough corresponde a una línea en el espacio de la imagen que pasa por estos dos puntos. Una vez identificadas las líneas lo que se hace es cubrirlas con líneas negras de un grosor mayor y así eliminarlas de la imagen.

Hasta este punto el algoritmo trabaja con el componente rojo de la imagen. Cabe destacar que a partir de este bloque los resultados del procesamiento de imágenes son aplicados a la capa de grises, por esta razón se puede apreciar en la figura 15e que la operación morfológica de erosión identifica algunas manchas que no estaban identificadas en la figura 15d.

La figura 15 muestra los resultados de aplicar el pre procesamiento en una imagen con manchas.

B. Procesamiento principal

Se convierte la imagen de RGB a escala de grises, para ello se emplea la suma ponderada de los componentes R, G y B utilizando (6)

$$Y = 0.299R + 0.587G + 0.114B \quad (6)$$

En la figura 16 se muestra el algoritmo propuesto para identificar manchas.

Se obtiene la imagen en grises y se le aplica un filtro Gaussiano para eliminar el ruido, después se binariza la imagen utilizando el método de Otsu, el cual permite obtener el umbral óptimo para cada imagen. Las líneas identificadas en el pre procesamiento se utilizan para eliminar la cuadrícula del panel solar.

El siguiente paso es aplicar una transformación morfológica. El objetivo de los operadores morfológicos es simplificar la

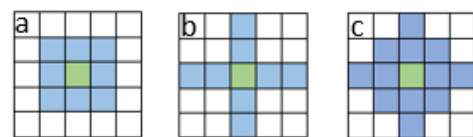


Figura 17. Elementos estructurantes: a) elemento estructurante cuadrado de 3x3, b) elemento estructurante cruz de 5x5, y c) elemento estructurante elipse de 5x5.



Figura 18. Ejemplo de erosión: a) imagen original, b) con un elemento estructurante de 3x3, y c) con un elemento estructurante de 7x7.

información de la imagen, conservar la forma esencial y eliminar ruido [11]. Para realizar las operaciones morfológicas se utilizan dos conjuntos, uno es la imagen (conjunto A) y el otro es el elemento estructurante (conjunto B). El conjunto B recorre al conjunto A haciendo un enmascaramiento [12]. El elemento estructurante tiene un origen, así como una forma y tamaño definidos. En la figura 17 se muestran algunos ejemplos de elementos estructurantes.

Los operadores morfológicos básicos son la erosión y la dilatación, de estos operadores surgen algunas combinaciones como son apertura y cierre. El resultado de aplicar una erosión en un conjunto A y B es el decremento de los píxeles. La erosión ocurre cuando el conjunto B recorre al conjunto A y los objetos menores al conjunto B desaparecen. Está definida por $A \ominus B$ donde $A \ominus B = \{s|(B)_z \subseteq A\}$. En la ecuación la erosión de A por B es el conjunto de todos los puntos z tales que B trasladado en z está contenido en A [11]. En la figura 18 se muestran algunos ejemplos de erosión. Este procedimiento es utilizado para eliminar ruido blanco o separar objetos conectados en la imagen.

En este punto se tiene una imagen limpia, únicamente con las manchas resaltadas. Se encuentran los contornos y se calcula su área, alto y ancho. Un contorno puede definirse como una curva que une todos los puntos continuos a lo largo del límite, que tienen el mismo color o intensidad. En el caso del algoritmo se busca un contorno blanco en un fondo negro. Una vez encontrados los contornos, se descartan aquellos que no cumplan con un rango de áreas permitidas, calculado a partir del histórico de manchas encontradas. Contornos demasiado grandes se descartan, así como contornos diminutos. La proporción entre alto y ancho también es un factor, dado que las manchas tienden a ser circulares o no demasiado alargadas. Los contornos que cumplan los parámetros son proyectados sobre la imagen original. En la figura 19 se muestra el resultado de aplicar el procedimiento principal.

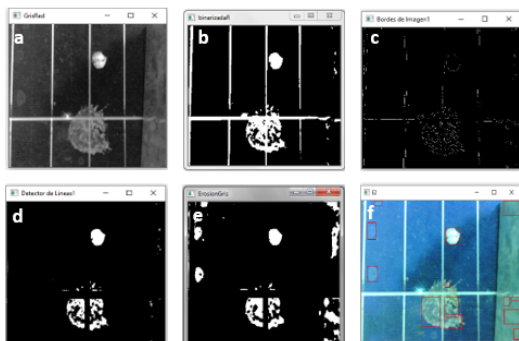


Figura 19. Resultados del procedimiento principal: a) imagen del canal rojo, b) binarizada, c) detección de bordes, d) detección y eliminación de líneas, e) erosión, y f) dibujar los contornos.

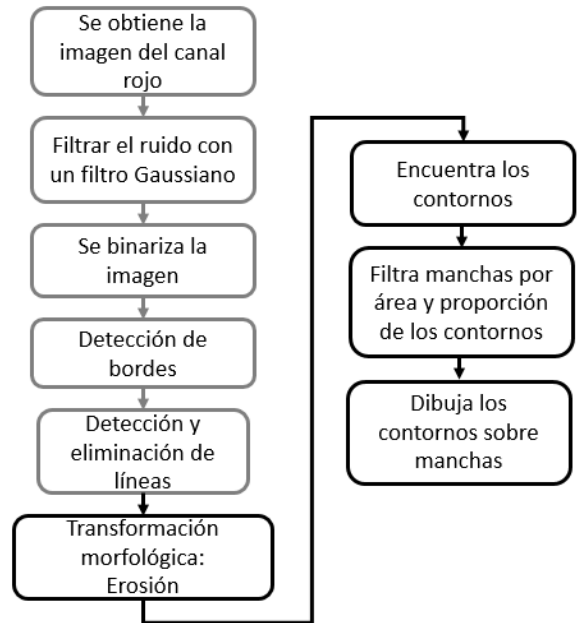


Figura 20. Algoritmo del procedimiento Rojo.

C. Procedimiento en Rojo

Se realizó un segundo procedimiento [Fig. 20], para identificar las manchas en imágenes que tuvieran condiciones diferentes de iluminación y que no hubieran sido identificadas por el procedimiento principal. Este segundo procedimiento es muy similar al anterior, su diferencia radica en que todo el procedimiento de principio a fin se aplica utilizando únicamente el componente rojo de la imagen, no la imagen en grises calculada al ponderar todos los componentes RGB, como en el caso anterior.

Se parte, entonces, de la imagen resultante del pre procesamiento y se aplican los últimos bloques, donde se realiza la operación morfológica de erosión, detección de contornos y filtrado por área y proporción, todo sobre el componente rojo.

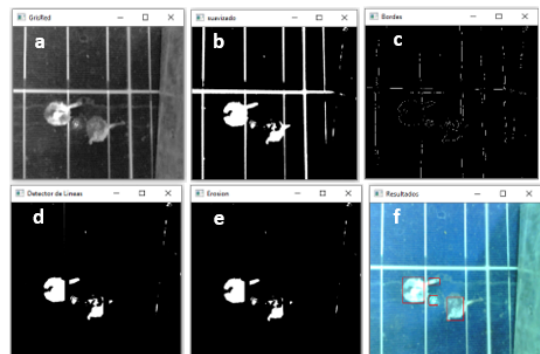


Figura 21. Resultados del Procedimiento en Rojo: a) imagen del canal rojo, b) binarizada, c) detección de bordes, d) detección y eliminación de líneas, e) erosión, f) dibujar los contornos.

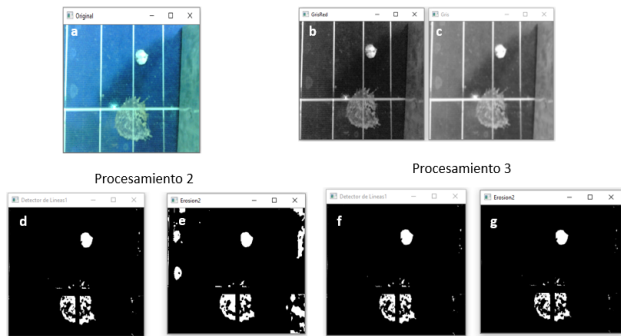


Figura 22. Diferencias entre procedimiento gris y procedimiento rojo.

Esta modificación del procedimiento rojo permite identificar manchas más tenues. En la figura 21 se muestran los resultados de aplicar el algoritmo de procedimiento en rojo a una imagen.

D. Diferencias entre Procedimiento Principal y Procedimiento en Rojo

Aunque pudieran parecer iguales el procedimiento principal y el procedimiento en rojo funcionan correctamente para diferentes manchas en diferentes tipos de imágenes. Manchas que no son identificadas por el procedimiento principal pueden ser identificadas por el procedimiento en rojo. Ya que el procedimiento en rojo se caracteriza por identificar las manchas aun variando las intensidades de iluminación en las imágenes. Mientras que el procedimiento principal es sensible a los cambios de iluminación así que puede identificar manchas que son sutiles o poco visibles.

La figura 22 muestra un ejemplo de una imagen analizada por los dos procedimientos. En la figura 22a se puede observar la imagen original del panel solar que contiene la mancha. En las figuras 22b y 22c se aprecia la mancha en los canales rojo y gris. Las figuras 22d y 22e muestran los resultados obtenidos del procedimiento en gris donde se puede apreciar que el procedimiento identifica manchas pocos visibles mientras que las figuras 22f y 22g se observan los resultados del procedimiento en rojo.

V. RESULTADOS

Con el clasificador se obtuvo un porcentaje de aciertos del 99.16 % [Fig. 23a] para el kernel lineal y de 95.58 % para el kernel radial [Fig. 23b]. Con el fin de ejemplificar de forma gráfica el proceso, en la figura 23 se presenta este con dos características, los valores del componente rojo y el componente verde y en esta gráfica en dos dimensiones se aprecia la separación entre las clases. También se probó con el algoritmo de K vecinos más cercanos. Para la comparación entre los algoritmos, los datos se dividieron 70 % para entrenamiento y 30 % para pruebas [Tabla 1].

TABLA I COMPARACIÓN ENTRE CLASIFICADORES

Comparación de algoritmos (% de aciertos)		
K vecinos más cercanos	Máquinas de soporte vectorial lineal	Máquinas de soporte vectorial radial
90%	99.16%	95.8%

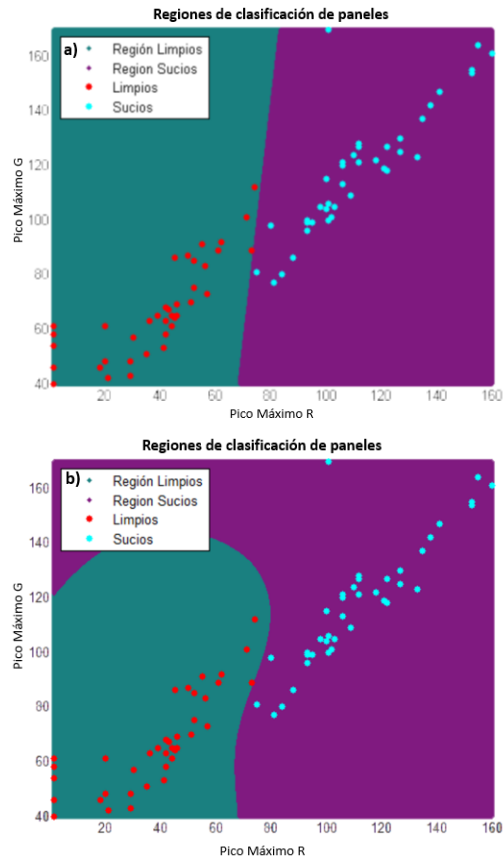


Figura 23. Clasificación con kernel lineal y base radial.

Para probar la eficiencia de los procedimientos propuestos se eligieron 50 imágenes de los paneles solares al azar. El procedimiento principal logró detectar e identificar en 29 imágenes (58%) el total de las manchas, en 15 imágenes (30%) identificó algunas manchas y solo en 8 imágenes (16%) no identificó manchas cuando sí las había. Mientras que el procedimiento en rojo identificó todas las manchas en 28 imágenes (56%), en 17 imágenes (34%) identificó algunas manchas y en 6 imágenes (12%) no identificó manchas. En total con el sistema de visión artificial se identificaron en 40 imágenes (80%) todas o algunas manchas y en 10 imágenes (20%) que había manchas no fueron identificadas ninguna. Al sistema de visión artificial le lleva 0.578 s el proceso fragmentar las imágenes, procesarlas y obtener los resultados. Por lo tanto, se considera que es factible la utilización de este sistema en la limpieza de paneles solares. El tiempo total que le tomé la limpieza de un solo panel solar completo dependerá de la velocidad de la plataforma móvil.

Estos algoritmos se están implementando en una tarjeta de desarrollo NVIDIA Jetson Nano, con el propósito de realizar el procesamiento de visión en tiempo real a bordo de la plataforma base descrita anteriormente, para la limpieza de paneles solares. Esta tarjeta cuenta con un GPU 128-core Maxwell™, CPU Quad-core ARM A57, 4 GB 64-bit LPDDR4 | 25.6 GB/s. Jetson Nano también es compatible con NVIDIA JetPack™, que incluye un paquete de soporte de placa (BSP), sistema operativo Linux, bibliotecas de software NVIDIA CUDA®, cuDNN y TensorRT™ para deep learning, visión por computadora,

computación GPU, lo que la hace ideal para realizar el procesamiento de imágenes propuesto.

VI. CONCLUSIONES

Los algoritmos descritos en este trabajo resultan efectivos para la tarea de clasificación de paneles limpios o sucios, así como para identificar manchas en paneles solares con el fin aplicar estrategias de limpieza.

Dependiendo de las condiciones de iluminación en las que se realicen las lecturas, podría ser más adecuado uno u otro de los procedimientos aquí descritos y en su caso, elegir utilizar solo uno de ellos para reducir los tiempos de proceso y poder tener una tasa de muestreo mayor.

Las pruebas aquí descritas se realizaron sobre imágenes estáticas, si se trabaja sobre una plataforma móvil que realice capturas de imágenes desde diferentes ángulos, puede favorecer a la identificación de manchas debido a que se tiene más de una oportunidad de detección, considerando que se estará realizando capturas de imágenes de forma continua.

Las tarjetas de desarrollo actuales como la NVIDIA Jetson nano, o equivalentes como la Raspberry Pi 4, poseen capacidad de cómputo suficiente para utilizar herramientas de programación como Python y OpenCV, lo que simplifica la implementación de algoritmos de visión artificial, además de incluir librerías optimizadas para una ejecución rápida, ideales para tareas en tiempo real.

AGRADECIMIENTOS

Agradecemos al Instituto Tecnológico de la Paz, y al Centro de Investigaciones Biológicas del Noroeste por el apoyo recibido para la realización de este trabajo. Gracias al M.S.C. Juan Francisco Villa Medina por sus consejos y aportaciones, así como a Jorge Cobos Anaya del taller de maquinados del

CIBNOR por su apoyo en la fabricación de la plataforma móvil y diferentes piezas utilizadas en este trabajo.

REFERENCIAS

- [1] Aristizábal Cardona Andrés Julián, Páez Chica Carlos Arturo, Ospina Barragán Daniel Hernán. "Building-Integrated Photovoltaic Systems (BIPVS) Performance and Modeling Under Outdoor". Conditions. Springer International Publishing, 2018
- [2] G Aravind, Gautham Vasani, T S.B.Gowtham Kumar, R Naresh Balaji, and G Saravana Ilango. "A Control Strategy for an Autonomous Robotic Vacuum Cleaner for Solar Panels". Proceedings - 2014 Texas Instruments India Educators Conference, TIEEC 2014, pages 53-61.
- [3] Mohamed Cherif Aidara, M. Ndiaye, A. Mbaye, M. Sylla, P. Ndiaye, A. Ndiaye less. "Study of the performance of a system for dry cleaning dust deposited on the surface of solar photovoltaic panels". International Journal of Physical Sciences International Journal of Physical Sciences Vol. 13(2), pp. 16-23, 30 January, 2018
- [4] Sitio web oficial de Robot Eccoppia URL: <https://www.ecoppia.com/technology>
- [5] K Matthias The development of a cleaning robot for PV panels. Ku Leuven 2014-2015
- [6] Thanapon Sorndach, Noppadol Pudchuen, and Pornsak Srisungsitthisunti. Rooftop Solar Panel Cleaning Robot Using Omni Wheels. 2018 2nd International Conference on Engineering Innovation, ICEI 2018, pages 7-12, 2018.
- [7] L Enrique Sucar and Giovanni Gómez. Vision Computacional. Instituto Nacional de Astrofísica, Óptica y Electrónica, page 185, 2015
- [8] Gutierrez, R. M., Rodriguez, C., & Triana, N. Técnicas de umbralización para el procesamiento digital de imágenes de GEM-Foils. Scientia et Technica Año Año XXI, Vol. 21, No. 4. 2016. Pages 352-359
- [9] Szeliski. Computer Vision. Algorithms and Applications. Springer-Verlag, 2011,
- [10] Rojas Z, Teddy V.; Sanz F., Wilmer; Arteaga, Francisco Sistema de visión por computadora para la detección de objetos esféricos a través de la transformada de Hough Revista Ingeniería UC, vol. 15, núm. 1, abril, 2008, pp. 77-87
- [11] Rafael C.; Woods Richard E. Gonzalez. Digital image processing, volume 8. Prentice Hall, 1990.
- [12] M. Goyal, "Morphological image processing," International Journal of Computer Science and Technology, vol. 2, no. 4, pp. 161-165, 2011

Sistema de percepción no centralizado para enjambres de robots RAOI basado en Deep Learning.

Erik Ricardo Palacios Garza
Facultad de Ingeniería Mecánica y Eléctrica
Universidad Autónoma de Nuevo León
San Nicolás de los Garza, N.L.
erik.palaciosgrz@uanl.edu.mx

Luis Martín Torres Treviño
Facultad de Ingeniería Mecánica y Eléctrica
Universidad Autónoma de Nuevo León
San Nicolás de los Garza, N.L.
luis.torrestrv@uanl.edu.mx

Resumen—En el área de la robótica la implementación de sistemas de percepción es de gran importancia, actualmente con el desarrollo del deep learning se han podido desarrollar sistemas de visión sumamente complejos, sin embargo, para el área de la robótica de enjambres esta tecnología aún no se ha adaptado a la percepción de cada integrante de los enjambres. Con este trabajo se ha implementado un sistema de visión basado en deep learning a un miembro de un enjambre de robots de configuración diferencial, esta implementación se realizó en una tarjeta raspberry pi utilizando su cámara correspondiente. Los resultados fueron el aumento en la cantidad de estímulos que el enjambre puede detectar, permitiendo el desarrollo de comportamientos más diversos.

Palabras clave:—Robótica de Enjambres, Deep Learning, Percepción, Visión Artificial.

I. INTRODUCCIÓN

Los sistemas de percepción han sido utilizados en robots para generar tareas cada vez más complejas en diferentes áreas de la ciencia, por ejemplo, en la medicina se aplica en robots quirúrgicos o mecanismos para la realización de diferentes estudios médicos, en la industria se utiliza en robots controladores de calidad, robots ensambladores entre otros, también dentro de la agricultura, con su implementación en robots controladores de plagas, sin embargo, en el área de robótica de enjambres los sistemas de percepción han sido limitados a sensores sencillos como ultrasónicos, infrarrojos de distancia, por mencionar algunos, limitando no solo la percepción de cada robot si no también el desarrollo de comportamientos nuevos.

Los sistemas de percepción han pasado por una gran evolución en los últimos años con la implementación de modelos de redes neuronales en el reconocimiento, clasificación y detección de objetos en imágenes, gracias a esto se ha ampliado el campo de aplicaciones con sistemas de visión, ya que anteriormente las técnicas de procesamiento de imágenes estaban limitadas a condiciones del ambiente en el cual se capturaban las imágenes, por lo que la complejidad del diseño de un sistema de percepción para enjambres de robots basado en procesamiento de imágenes se vuelve altamente complejo, una posible adaptación de estas técnicas es la utilización

de modelos centralizados, sin embargo en la robótica de enjambres limita la movilidad y la diversidad de aplicaciones que se pueden realizar.

II. ANTECEDENTES

La robótica de enjambres trata de resolver problemas complejos usando un conjunto de robots de baja complejidad y bajo costo, como por ejemplo exploración, localización, construcción etc. Una de las ventajas de trabajar con estas configuraciones es que pueden cumplir con la tarea asignada aun teniendo la pérdida o inclusión de miembros nuevos al enjambre, así como disminuir la complejidad de los algoritmos necesarios para la realización de diversas tareas, al emerger comportamientos propios del enjambre.

En la mayoría de los casos, como se menciona en [1] estos robots son construidos con un cierto nivel de percepción, el cual consiste en un conjunto de sensores que les permiten tener cierta información del espacio que los rodea, los obstáculos que se podrían encontrar, u objetos con los cuales interactuar, por ejemplo, los robots utilizados en [1] utilizan sensores de proximidad y luz para medir la distancia con robots vecinos u objetos, además de obtener información con la luz del lugar donde se encuentren, con esta percepción asignada a cada robot se han obtenido resultados prometedores como los mencionados en el artículo, sin embargo se tiene la limitante de solo obtener información local de medio ambiente, como la que ya se mencionó.

En [2] se menciona una problemática de percepción similar que en [1] donde la comunicación a corta distancia, representa un reto que propicia el desarrollo de algoritmos más complejos para la robótica de enjambres, presentando la percepción asignada a cada robot como restrictiva.

Existen algunos otros sistemas de percepción para robótica de enjambres más complejos que los sensores antes mencionados que a su vez proporcionan una mayor capacidad de trabajo a cada robot y al sistema completo, como es el caso de [3], donde utilizan gestos de manos para manipular el comportamiento grupal de los robots, este reconocimiento de gestos manuales está basado en un preprocesado de una

imagen de una mano en alguna posición específica, una extracción de características y un clasificador basados en SVM's. Claramente agregando un sistema de visión se puede observar que se tienen comportamientos que considerando un sistema de visión por sensores serían complicados de obtener.

En [4] se plantea resolver la tarea anterior con la utilización de redes neuronales convolucionales (CNN, por sus siglas en inglés), donde dichas redes presentan la ventaja de tener una alta precisión en la tarea de visión que se les asigne, además poseer gran robustez a las condiciones ambientales donde se capture la imagen a analizar, en la actualidad existe equipo en donde se pueden embeber y utilizar en sistemas robóticos móviles, la desventaja ahora presente es el poder de cómputo que se necesita para su entrenamiento, donde es necesario tanto un gran set de imágenes así como computadoras suficientemente potentes para ejecutar el entrenamiento que requieren.

Las CNN buscan mascarar óptimas para la extracción de características específicas de un objeto y así poder hacer su reconocimiento, clasificación o detección [5], lo cual por el contrario en el procesamiento de imágenes se realiza de manera manual, comúnmente las CNN constan de una parte donde se utilizan "convoluciones" con las máscaras antes mencionadas y una parte densa que proporciona una salida de clasificación [6], siendo esta configuración una de las más utilizadas.

Regresando a [4] se propone un modelo de CNN donde cambian la red de perceptrón por una SVM, dando como resultado una mejora en la precisión de la CNN. Con esta implementación en un enjambre de robots se podrían abrir una serie de tareas y comportamientos más complejos o mejorar las tareas que ya se realizan en la actualidad. Sin embargo, si en lugar de implementar una CNN entrenada para detectar gestos realizados con las manos, se entrenan para poder percibir su medio ambiente, se podría dar lugar a comportamientos más eficientes y más complejos que podrá realizar el enjambre sin la necesidad de intervención humana como antes se menciona, un ejemplo de las ventajas de estas nuevas alternativas a estos comportamientos es la solución a una problemática planteada en [7] donde se menciona que con sensores como los antes mencionados un robot no tiene la posibilidad de conocer la orientación de sus vecinos, con la implementación de una CNN se puede abrir esta capacidad a los robots.

III. CONTRIBUCIÓN

En este trabajo se presenta un sistema de percepción no centralizado aplicable a enjambres de robots, el cual cumple con la función de detectar diversos estímulos de influencia y objetos de interés dentro de las zonas de trabajo de los enjambres propuestas en [15], con el aumento drástico en los estímulos de influencia del enjambre este puede generar nuevos y más complejos comportamientos.

El resto del documento se organiza de la siguiente manera: En la siguiente sección se explican los modelos de Deep learning utilizados para la detección correcta de los estímulos de influencia, así como conceptos básicos de enjambres de robots RAOI. Luego, se explica el hardware y desarrollo

del prototipo robótico de experimentación. Posteriormente, se ilustran los resultados de experimentos de seguimiento de un estímulo de influencia en cuatro escenarios de prueba, finalmente se presenta la discusión sobre los resultados y conclusiones.

IV. ENJAMBRES DE ROBOTS RAOI

El sistema de percepción que se propone está enfocado para el desarrollo de comportamientos para enjambres de robots RAOI propuestos en Couzin [15], este tipo de comportamientos están basados en características de animales sociales, por ejemplo, abejas, hormigas, lobos, entre otros. Donde se consideran tres características básicas de su movimiento para la generación comportamientos parecidos a los de ellos.

Estas características constan de áreas en las cuales cada animal acostumbra acercarse, alejarse u orientarse en la misma dirección que sus vecinos, estas fueron nombradas zonas de repulsión, orientación y atracción [9], se pueden observar en la figura 1, dentro de la primera zona el individuo trata de alejarse de sus vecinos ya que los considera demasiado cercanos ya sea para realizar algún movimiento o por una posible colisión, la segunda zona es cuando el individuo considera tener una distancia adecuada de sus vecinos orientándose a la misma dirección que ellos, en la última zona si el individuo se encuentra muy distanciado de sus vecinos este tenderá a acercarse a sus compañeros para integrarse al enjambre.

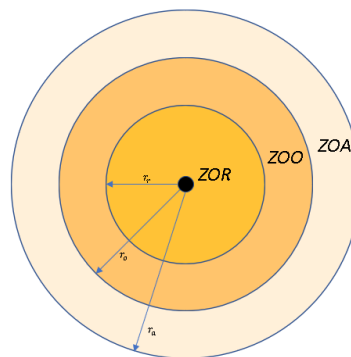


Fig. 1. Zonas de repulsión, orientación y atracción

Basándose en estas tres zonas se pueden generar comportamientos similares a los que realizan los animales mencionados anteriormente, sin embargo, en [10] se agregó una nueva zona llamada de influencia la cual se puede apreciar en la figura 2, donde si se tiene la existencia de algún estímulo para el enjambre, tenderá a realizar la tarea que el estímulo indique, sin embargo como ya se mencionó antes debido a las limitaciones que se tienen en los sistemas de percepción de los enjambres, el número y tipo de estímulos de influencia se limitaba a las capacidades de los sensores simples que utilizaban.

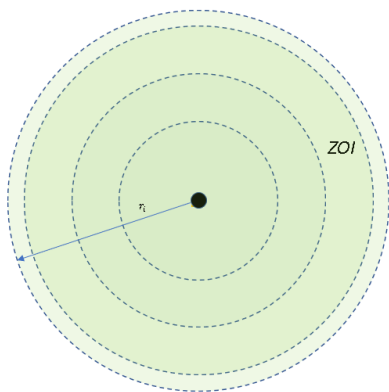


Fig. 2. Zona de influencia

V. MODELOS DE DEEP LEARNING BASADOS EN REDES NEURONALES CONVOLUCIONALES.

A. Redes neuronales convolucionales

En el área de visión artificial se trata de cumplir con diferentes objetivos teniendo como principales el reconocimiento, la clasificación y la detección de objetos, estos objetivos se pueden cumplir con la ayuda de redes neuronales, las cuales aprenden a extraer características específicas de una imagen para realizar tarea que se desea cumplir, estas redes constan principalmente de los siguientes elementos:

- *Capas convolucionales:* Constan de un mapeo de una matriz de pesos o filtro a lo largo y ancho de una determinada imagen, haciendo la suma de las multiplicaciones los pesos del filtro por los correspondientes píxeles de la imagen, esta operación está dada por la expresión 1.

$$C = f(I * K + b) \quad (1)$$

$$C(i, j) = f \left(\sum_{u=0}^{Fi} \sum_{v=0}^{Co} I(i+u, j+v) \cdot K(u, v) + b \right) \quad (2)$$

- *Capas de pooling:* Esta capa redimensiona las salidas de las capas convolucionales para facilitar su procesamiento por capas posteriores a esta operación, existen diferentes esquemas de pooling, a continuación, se presenta uno de los más comunes utilizado, el cual divide la imagen original en grupos de 4 píxeles vecinos, obtiene el promedio de estos cuatro píxeles y los anexa al píxel correspondiente a la vecindad elegida en la imagen original.

$$S(i, j) = \frac{1}{4} \sum_{u=0}^1 \sum_{v=0}^1 C(i+u, j+v) \quad (3)$$

- *Padding:* Consiste en agregar un marco de ceros a la imagen de entrada, sirve para realizar una convolución completa a la imagen.

- *Vectorización:* Este proceso consiste en el redimensionamiento de la última capa convolución a un vector para poder ser procesado por la parte densa de la red.

$$v(k) = C(i, j) \quad (4)$$

Para $i = 0, 1, 2, \dots, Fi$, $j = 0, 1, 2, \dots, Co$

B. Redes neuronales residuales

Durante el proceso de convolución se tiene una pérdida de información al momento de tratar de incrementar el número de capas para mejorar la eficiencia de la red, esta pérdida de información se puede compensar con la utilización de conexiones residuales, las cuales realizan la integración de información de la entrada de una determinada capa a la salida de otra permitiendo extender la profundidad de capas sin pérdida de porcentaje en métricas de evaluación, el conjunto de todas las capas que salta la conexión residual se le llama bloque residual y al conjunto de bloques residuales se le conoce como red residual o Resnet, en la figura 3 se puede apreciar un bloque residual.

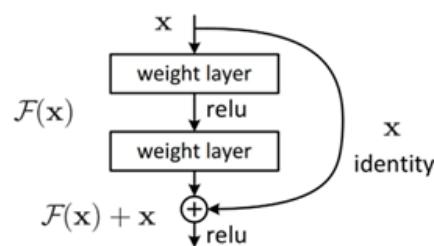


Fig. 3. Modelo básico de un bloque residual [11]

C. Redes de detección

La limitación de usar redes de clasificación es que no se puede identificar la posición del objeto en la imagen lo cual es de gran utilidad para el direccionamiento de los robots, existen diferentes modelos de detección en este caso se utilizará el modelo de YOLO (you only look once) [12] en su versión tiny propuesta en Rachel Huang [14] para poder obtener la máxima velocidad de respuesta posible en cada robot.

Esta red se basa en dividir la imagen a analizar en celdas, donde cada celda dará las coordenadas, ancho y largo de una serie de encuadres también llamados bounding boxes, los cuales trataran de encerrar el objeto de interés de manera precisa, también cada celda otorga información acerca de la posible existencia de un objeto de interés en la celda y la clase a la cual puede pertenecer el objeto.

D. Robots diferenciales

La configuración de robot móvil que se utilizó fue la diferencial, teniendo su modelo matemático basado en A. Bara [13] a continuación:

- Modelo cinemático.

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos(\theta) - \frac{rd}{2R} \sin(\theta) & \frac{r}{2} \cos(\theta) + \frac{rd}{2R} \sin(\theta) \\ \frac{r}{2} \sin(\theta) + \frac{rd}{2R} \cos(\theta) & \frac{r}{2} \sin(\theta) - \frac{rd}{2R} \cos(\theta) \\ \frac{r}{2R} & -\frac{r}{2R} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (5)$$

- Modelo Dinámico.

$$\mathbf{M} \cdot \dot{v} + H(v) = \mathbf{B} \cdot \tau \quad (6)$$

donde:

$$\begin{bmatrix} m & 0 \\ 0 & I_p + m \cdot d^2 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} -m \cdot d \cdot \dot{\theta}^2 \\ m \cdot d \cdot v \cdot \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{R} & \frac{1}{r} \\ \frac{r}{R} & -\frac{r}{R} \end{bmatrix} \begin{bmatrix} \tau_d r \\ \tau_d l \end{bmatrix} \quad (7)$$

Considerando motores de CD el modelo completo está dado por:

$$(M + BK_L^{-1}CA^{-1})\dot{\mathbf{v}} + (H(\mathbf{v}) + BK_L^{-1}A^{-1}\mathbf{v}) = BK_L^{-1}K_{in}u(8)$$

donde $u = [u_r \ u_l]^T$ es el voltaje de entrada a cada motor.

$$C = \begin{bmatrix} c_r & 0 \\ 0 & c_l \end{bmatrix}, K_L = \begin{bmatrix} k_{L,r} & 0 \\ 0 & k_{L,l} \end{bmatrix},$$

$$K_{in} = \begin{bmatrix} k_{in,r} & 0 \\ 0 & k_{in,l} \end{bmatrix}, A = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{2R} & -\frac{r}{2R} \end{bmatrix}$$

VI. PROPUESTA

A. Comportamiento de experimentación

Para poner a prueba el sistema de percepción propuesto, se utilizó un comportamiento de búsqueda aplicado como una máquina de estados finita ilustrada en la figura 4, compuesta por tres estados, “búsqueda”, “atención” y “evasión”.

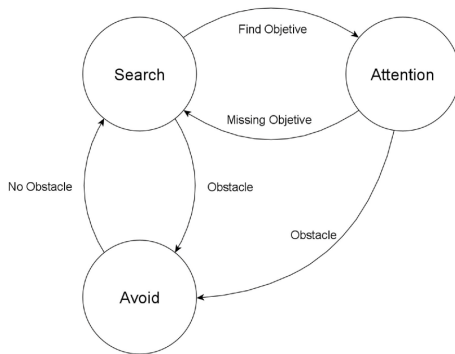


Fig. 4. Comportamiento experimental

Donde el estado de búsqueda se encargará de encontrar un estímulo de influencia dentro de la imagen captada por el robot, el estado de atención encuentra la orientación del estímulo de influencia con respecto al robot así como una aproximación de la distancia entre estos dos, por último el estado de evasión se encarga de evitar obstáculos mediante sensores infrarrojos.

B. Esquema propuesto de un sistema de percepción para enjambres de robots.

Teniendo las herramientas del Deep Learning que se mencionaron anteriormente, se pueden utilizar en conjunto para poder cumplir con las demandas de percepción que tiene cada robot, por lo que se propone un sistema compuesto por dos redes neuronales, una de clasificación y una red de detección, esto disminuye el coste computacional en la tarjeta utilizada, ya que el tiempo de ejecución de la red de detección es muy alto se propone reducir su tiempo activo, por lo que para fines del comportamiento que se desea reproducir, estará activa la mayor parte del tiempo la red de clasificación y solo en cuanto se detecte una señal de influencia del enjambre se activará la red de detección como se puede observar en la figura 5.

La red de clasificación consiste en una red residual de tres bloques como el que se aprecia en la figura 6, se llegó a este bloque residual partiendo de configuraciones sencillas donde algunas demostraban tener una buena precisión pero un tiempo de ejecución muy largo para la tarea planeada o en casos contrarios tiempos de ejecución adecuados pero precisiones bajas, por lo que después de varias pruebas se llegó a esta configuración obteniendo ambas cualidades dentro de rangos aceptables, la configuración de cada bloque se puede apreciar en la figura 6.

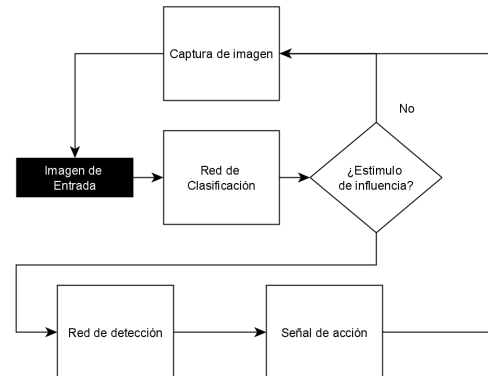


Fig. 5. Esquema de percepción propuesto

En cuanto a la red de detección como ya se mencionó en la sección anterior se utilizará la red YOLO en su versión tiny, esto por su precisión y velocidad demostradas en la literatura.

C. Prototipo experimental

Para poner a prueba el sistema de percepción se sometió a un comportamiento de búsqueda un solo robot como el que se aprecia en la figura 8

El hardware del robot se describe a continuación:

- *Microcontrolador*: Para embeber el algoritmo del controlador de los motores y del sistema de percepción se utilizó la tarjeta electrónica *Raspberry pi*® 3 B. Las características más importantes para la selección de esta tarjeta son principalmente, la capacidad de procesamiento y el espacio de memoria disponible para almacenamiento

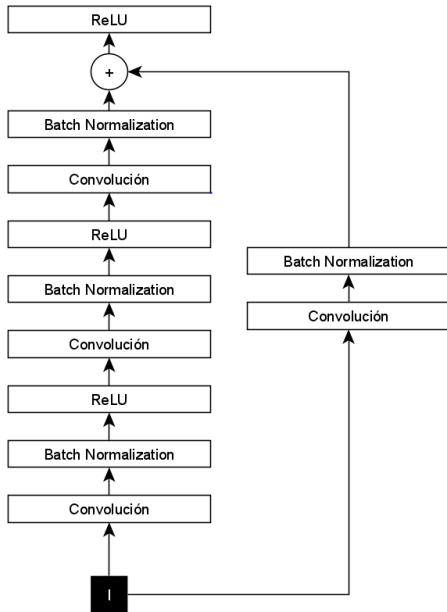


Fig. 6. Bloque residual propuesto.

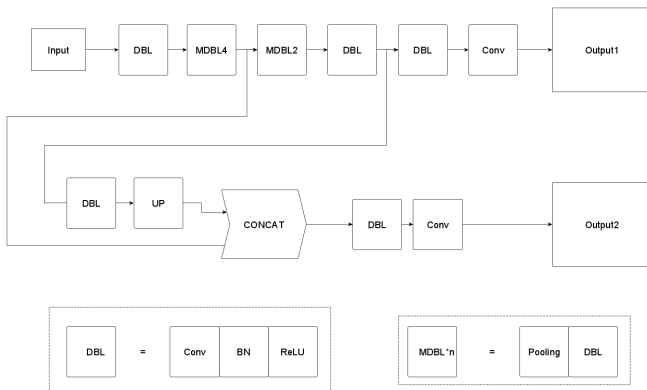


Fig. 7. Red YOLO Tiny

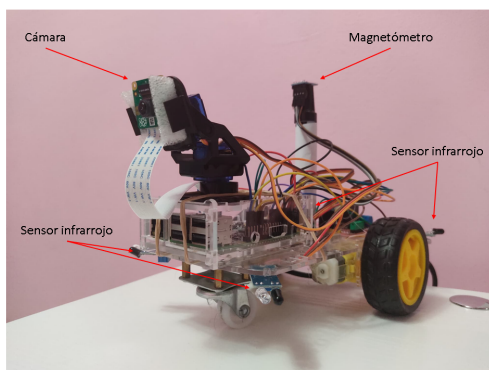


Fig. 8. Prototipo experimental

de código y datos además de mantener en la medida de lo posible el rasgo minimalista del hardware de los robots.

- *Motores de CD*: El robot tiene dos motores de corriente directa controlados con controladores PD.
- *Magnetómetro*: Debido a que en el momento de orientar al robot esta activa la red de detección no se tiene una retroalimentación lo suficientemente veloz del posicionamiento del estímulo de influencia, por lo que el objetivo del magnetómetro es proporcionar una retroalimentación de error angular o de orientación que se tiene del robot con respecto al objeto de interés.
- *Sensores infrarrojos*: Se utilizan para evitar posibles obstáculos.

Las pruebas de experimentación se realizaron en un ambiente de dimensiones controladas donde se colocó un objetivo en 4 posiciones distintas.

D. Control

Para lograr el control correcto de los motores se utilizaron dos controladores PD, uno controla la orientación del robot utilizando como señal de retroalimentación el magnetómetro, realizando una relación lineal entre el ángulo de visión de la cámara y el ancho de la imagen capturada, como se puede apreciar en la figura 9 utilizando la información de la bounding box del estímulo de influencia se calcula C_o y utilizando la relación ya mencionada se obtienen los grados que se debe de girar para empatar el centro de la imagen con el centro de la bounding box.

El segundo controlador se encarga de la distancia que hay entre el robot y el estímulo de influencia. Debido a la variabilidad de las mediciones realizadas por un sensor ultrasónico causadas por el corto ángulo de amplitud que tiene, no es viable usarlo como retroalimentación, en su lugar se utiliza el ancho de la bounding box del estímulo de influencia, esta se puede observar cómo W_o en la figura 9

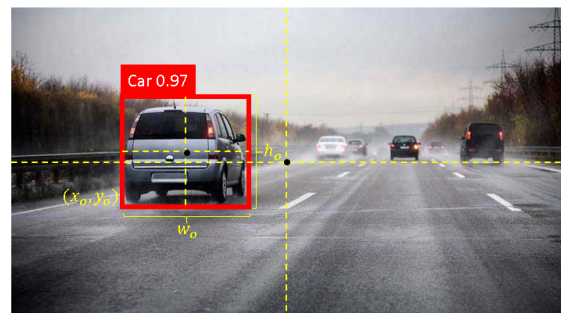


Fig. 9. Obtención de magnitudes

VII. RESULTADOS

A. Entrenamiento de red de clasificación.

El entrenamiento se realizó con 600 imágenes de 3 diferentes clases "Vacío", "Obstáculo" y "Objetivo" a 100 épocas, en las figuras 10 y 11 se puede apreciar la evolución del error y la precisión de la red de clasificación a lo largo del entrenamiento.

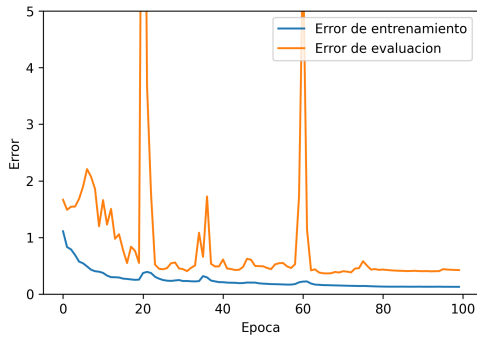


Fig. 10. Evolución del error en la red de clasificación

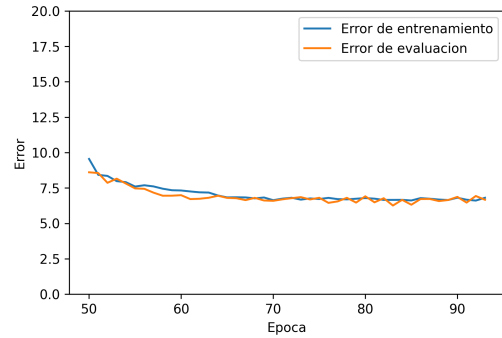


Fig. 13. Evolución de la precisión en la red de detección (épocas 50-100)

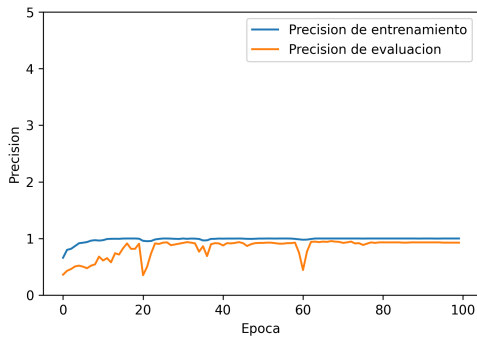


Fig. 11. Evolución de la precisión en la red de clasificación

B. Entrenamiento de red de detección.

El entrenamiento se realizó con 1000 imágenes de la categoría "Objetivo" a 100 épocas, en las figuras 12 y 13 se puede apreciar la evolución del error de la red de detección a lo largo del entrenamiento.

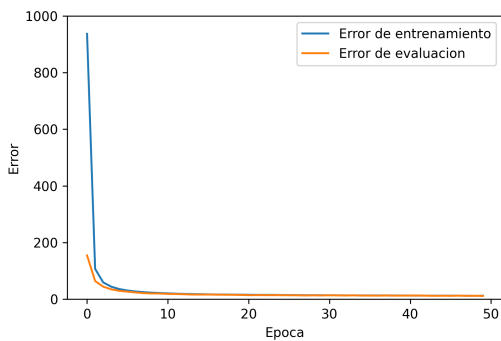


Fig. 12. Evolución del error en la red de detección (épocas 0-50)

C. Experimentación de comportamiento de búsqueda.

Durante los cuatro experimentos se tomaron diferentes posiciones y perspectivas del estímulo de influencia, además de diferentes iluminaciones. Donde se observó que el robot

pudo encontrar la fuente de influencia sin importar la presencia de diferentes posiciones y distancias con respecto al objeto.

1) *Estado de búsqueda y evasión.*: En la figura 14 se puede observar el estado de búsqueda del robot, donde se encuentra activa la red de reconocimiento la cual se encarga de decir si existe un objeto de interés en la imagen que está observando el robot, como no se aprecia el estímulo de influencia se mantiene en el estado de búsqueda, esta red tuvo un tiempo promedio de ejecución de 0.2 segundos,

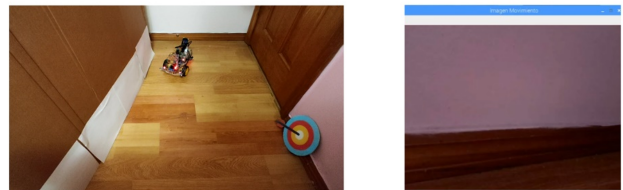


Fig. 14. Muestra de estado de búsqueda

2) *Estado de Atención:* Una vez que se localiza un estímulo de influencia se entra en el estado de atención y empieza a actuar la red de detección como se aprecia en la figura 15 donde se localiza el objeto, se ajusta la orientación del robot hacia el estímulo para a continuación aproximarse hacia él, deteniéndose hasta que el estímulo sea lo suficientemente cercano para no colisionar como se observa en la figura 16, esta red tuvo un tiempo promedio de ejecución de 0.7 segundos.

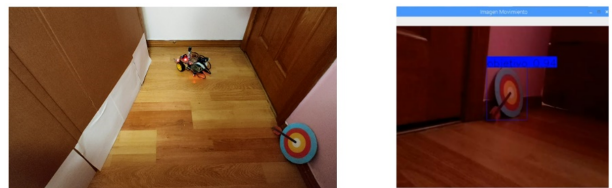


Fig. 15. Muestra de estado de atención

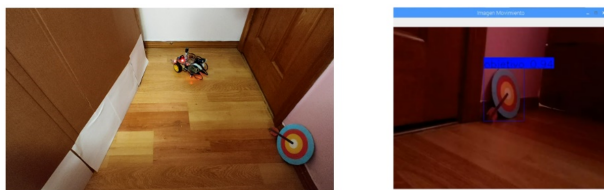


Fig. 16. Muestra de estado de atención (aproximación)

VIII. DISCUSIÓN DE RESULTADOS

Los experimentos muestran resultados aceptables cumpliendo con el objetivo del comportamiento de búsqueda, al tener una buena precisión en las redes de clasificación y detección se logra encontrar el estímulo de interés, sin embargo, al aumentar el número de estímulos de influencia en la red de clasificación incrementa la dificultad de generalizar el concepto de fondo dentro de la red, es decir lograr que la red pueda decir si existe o no un objeto de interés. Otros factores de ruido que se encontraron en la experimentación fueron la no linealidad del magnetómetro donde se tuvieron problemas en establecer correctamente la orientación real del estímulo de influencia, aun teniendo algunas señales de error como la que ya se mencionó además de una falta de estabilidad en las llantas, el robot logra aproximarse adecuadamente al objeto de interés.

IX. CONCLUSIONES

La implementación del sistema de visión propuesto mostró que es posible la localización de objetos en tiempo real, utilizando un enfoque descentralizado el cual puede ser utilizado para desarrollar nuevos comportamientos dentro de la robótica de enjambres, dejando de depender de sensores limitados a la medición de magnitudes básicas del entorno del robot, cabe mencionar que estos sensores siguen siendo indispensables para el funcionamiento correcto de un enjambre, al contribuir a la detección de señales del medio ambiente del robot que no serían claramente reconocidas por un sistema de visión convencional, como por ejemplo, cambios sutiles en iluminación o sonidos que pudieran ser de utilidad en el desarrollo de comportamientos en los enjambres de robots, además de poder demostrar que con la implementación de sistemas de visión basados en Deep Learning es posible obtener múltiples estímulos de influencia en el enjambre dando paso al desarrollo de comportamientos más diversos, además de notar que la utilización de una o más redes de clasificación a un sistema de visión de cualquier modalidad, permite la disminución de falsos positivos en esta tarea.

AGRADECIMIENTOS

Los autores agradecen al CONACYT por el financiamiento para esta investigación, así como a la Facultad de Ingeniería Mecánica y Eléctrica por facilitar los materiales y herramientas necesarias en esta investigación.

REFERENCES

- [1] Erick Ordaz-Rivas, Angel Rodríguez-Linan, and Luis Torres-Trevino. Collective behaviors in swarms of builder robots. *Research in Computing Science*, 148:103–114, 2019
- [2] Thomas Schmickl, Christoph M oslinger, and Karl Crailsheim. Collective perception in a robot swarm. In *International Workshop on Swarm Robotics*, pages 144–157. Springer, 2006
- [3] Alessandro Giusti, Jawad Nagi, Luca Gambardella, and Gianni A Di Caro. Cooperative sensing and recognition by a swarm of mobile robots. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 551–558. IEEE, 2012.
- [4] Jawad Nagi, Gianni A Di Caro, Alessandro Giusti, Farrukh Nagi, and Luca M Gambardella. Convolutional neural support vector machines: hybrid visual pattern classifiers for multi-robot systems. In *2012 11th International Conference on Machine Learning and Applications*, volume 1, pages 27–32. IEEE, 2012.
- [5] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, 2019.
- [6] Zhifei Zhang. Derivation of backpropagation in convolutional neural network (cnn). University of Tennessee, Knoxville, TN, 2016.
- [7] Erick Ordaz-Rivas, Angel Rodríguez-Li'nan, Mario Aguilera-Ru'iz, and Luis Torres-Trevi'no. Collective tasks for a flock of robots using influence factor. *Journal of Intelligent Robotic Systems*, 94(2):439–453, 2019.
- [8] Ian D. Couzin, Jens Krause, Richard James, Graeme D. Ruxton, and Nigel R. Franks. *Collective Memory and Spatial Sorting in Animal Groups*. Elsevier Science Ltd, 2002(218):1–11, 2002.
- [9] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *SIGGRAPH Comput. Graph.*, 21(4):25–34, August 1987.
- [10] Ordaz Rivas, Erick de Jesús. *Colaboración emergente en enjambres de robots, con reglas inspiradas en el comportamiento de animales sociales*. Diss. Universidad Autónoma de Nuevo León, 2020.
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for imagerecognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016
- [12] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016
- [13] Bara, A., Dale, S.: Dynamic modeling and stabilization of wheeled mobile robot. In: *Proceedings of the 5th WSEAS International Conference on Dynamical Systems and Control*. pp. 87–92. CONTROL'09, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA (2009), <http://dl.acm.org/citation.cfm?id=1628055.1628077>
- [14] Rachel Huang, Jonathan Podoem, and Cuixian Chen. Yolo-lite: a real-time object detection algorithm optimized for non-gpu computers. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 2503–2510. IEEE, 2018
- [15] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, "Collective memory and spatial sorting in animal groups," *Journal of theoretical biology*, vol. 218, no. 1, pp. 1–11, 2002

Sistema de Navegación de Bajo Costo Basado en Visión Artificial

Octavio Paz Geraldo Sánchez
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
 La Paz, México
 M19310013@lapaz.tecnm.mx

José Luis Gómez Torres
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
 La Paz, México
 jose.gt@lapaz.tecnm.mx

Israel Marcos Santillán Méndez
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
 La Paz, México
 israel.sm@lapaz.tecnm.mx

Resumen—En este trabajo se propone una plataforma de bajo costo para evaluación de algoritmos de navegación. La plataforma consiste en un robot diferencial, un sistema de visión monocular y un mapa con espacios libres y obstáculos generado a partir del sistema de visión. La captura y procesamiento de imágenes se realiza utilizando la librería OpenCV en el entorno de desarrollo de QtCreator.

Para controlar la posición y orientación del robot se implementa un control PI. Las señales de control de posición y orientación se normalizan entre -1 y 1 y son enviadas mediante un protocolo TCP, de forma inalámbrica con un microcontrolador ESP8266.

Palabras clave—Robot Móvil, Visión Artificial, Navegación

I. INTRODUCCIÓN

Un robot móvil deberá responder tres preguntas fundamentales: ¿dónde estoy?, ¿a dónde voy? y ¿cómo llego ahí?, para responder estas preguntas, un robot autónomo deberá tener un mapa del entorno, conocer su ubicación dentro del mismo, planificar y ejecutar el movimiento [1].

La autonomía de un robot móvil se basa en el sistema de navegación automática, que incluye planificación, percepción y control [2]. El desafío de navegación para un robot implica ejecutar un curso de acción (o plan) para alcanzar su posición objetivo. Durante la ejecución, el robot debe reaccionar ante eventos imprevistos (por ejemplo, obstáculos) de tal manera que aún alcance la meta. Sin reaccionar, el esfuerzo de planificación no dará sus frutos porque el robot nunca alcanzará físicamente su objetivo. Sin planificación, el esfuerzo de reacción no puede guiar el comportamiento general del robot para alcanzar un objetivo distante; de nuevo, el robot nunca alcanzará su objetivo [3].

La representación del entorno del robot puede variar desde una descripción geométrica continua hasta un mapa geométrico basado en la descomposición o incluso un mapa topológico. El primer paso de cualquier sistema de planificación de rutas es, por lo tanto, transformar un modelo ambiental posiblemente continuo en un mapa discreto adecuado para el algoritmo de planificación de ruta elegido [2].

Las técnicas de búsqueda de grafos han estado fuertemente arraigadas en el campo de las matemáticas. No obstante, en los últimos años gran parte de la innovación se ha diseñado en la comunidad de la robótica. Esto puede atribuirse en gran medida a la necesidad de algoritmos capaces de trabajar en tiempo real, que puedan adaptarse a mapas en evolución y, por

tanto, a grafos cambiantes. Para la mayoría de estos métodos, distinguimos dos pasos principales: construcción de grafos, donde los nodos se colocan y conectan a través de aristas, y búsqueda de grafos, donde se realiza el cálculo de una solución (óptima) [4].

Cualquiera que sea la representación del mapa que se elija, el objetivo de la planificación de la ruta es encontrar la mejor trayectoria entre el inicio y el destino, en función de los criterios de optimización seleccionados (por ejemplo, la ruta más corta) [5].

En la literatura se desarrollan algunos sistemas basados en visión artificial y utilizan A* [6]. En [3] desarrollan un sistema cableado utilizando ROS (robot operating system), a diferencia de [6] en este trabajo se propone una plataforma inalámbrica. En [7] se desarrolla un sistema de vehículo guiado por colores que utiliza un sistema mecatrónico, que incluye visión artificial, neumática, Arduino y módulo inalámbrico Zigbee. En [10] se implementa un sistema que reporta su localización utilizando un hardware de mínimas prestaciones y marcas artificiales en el terreno, en el artículo solamente abarcan la localización del robot. En [11] se diseña una estrategia de planeación de rutas para la navegación autónoma de robots en entornos interiores con base en el aprendizaje automático, para luego ser implementado en la plataforma robótica Turtlebot del simulador Gazebo.

En este trabajo se propone una plataforma de bajo costo para evaluación de algoritmos de navegación. La plataforma consiste en un robot diferencial, un sistema de visión monocular y un mapa con espacios libres y obstáculos generado a partir del sistema de visión. La captura y procesamiento de imágenes se realiza utilizando la librería OpenCV en el entorno de desarrollo de QtCreator.

Para controlar la posición y orientación del robot se implementa un control PI. Las señales de control de posición y orientación se normalizan entre -1 y 1 y son enviadas mediante un protocolo TCP, de forma inalámbrica con un microcontrolador ESP8266, este modulo resulta mas barato que utilizar un modulo zigbee como se utiliza en [7].

La estructura general del reporte es la siguiente. En la sección II se describe el sistema propuesto, las restricciones, los requerimientos de diseño así como los componentes principales del sistema. En la sección III se describe el

procesamiento de las imágenes que inicia con la captura de la imagen, para finalmente encontrar la posición y orientación del robot. En la sección IV se muestra el proceso para obtener un mapa y generar una ruta. En la sección V se muestra el seguimiento de la trayectoria, el control, envío y recepción de datos. En la sección VI se muestran los resultados y en la sección VII conclusiones y trabajo futuro.

II. SISTEMA PROPUESTO

II-A. Descripción

Un requerimiento de diseño fue que la plataforma sea de bajo costo y fácil de replicar. Los componentes se indican en Tabla I y Fig. 1.

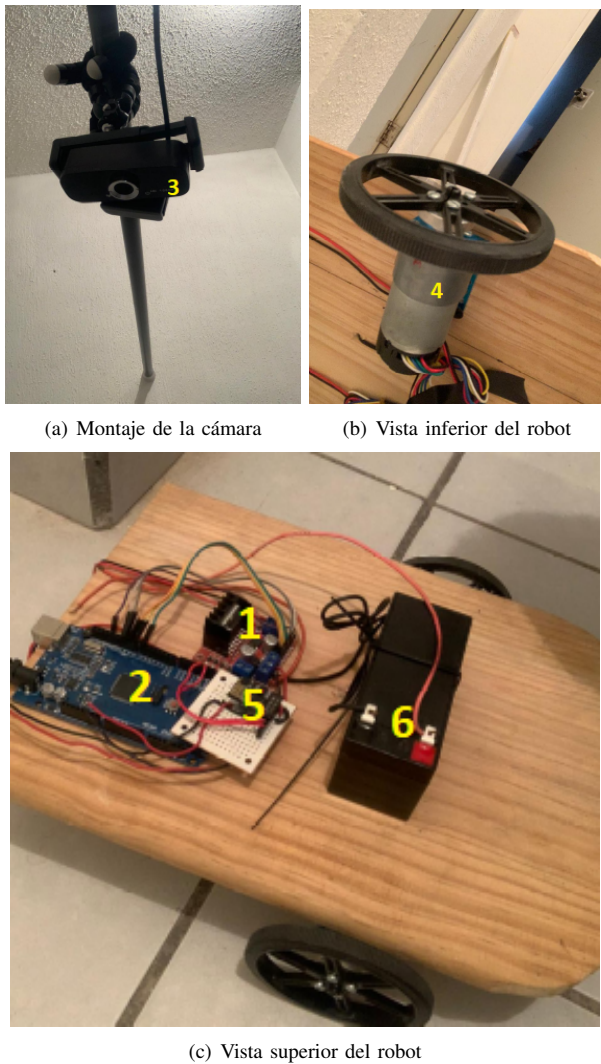


Figura 1. Componentes del sistema

En Fig. 2 se puede apreciar el diagrama general del sistema. Hay una cámara que capta el mapa completo, puede ver los obstáculos y al robot. Esta captura es procesada en una computadora que envía la señal de control al robot inalámbricamente mediante un protocolo TCP. En la Fig. 3 Se observa

Tabla I
COMPONENTES DEL SISTEMA

No.	Modelo	Componente
1	Puente H L298	
2	Arduino Mega 5260	
3	Webcam V3-1080P	
4	Motor Pololu 37D Metal Gearmotor 12V con buje de aluminio universal y ruedas de 90x10mm	
5	Modulo ESP8266	
6	Batería 12V 1.2 Ah	

un panorama completo del sistema. La cámara esta ubicada a dos metros de altura y abarca un área de 1.5m x 2.7m. sin embargo solo se utiliza un área de 0.9m x 1.5m.

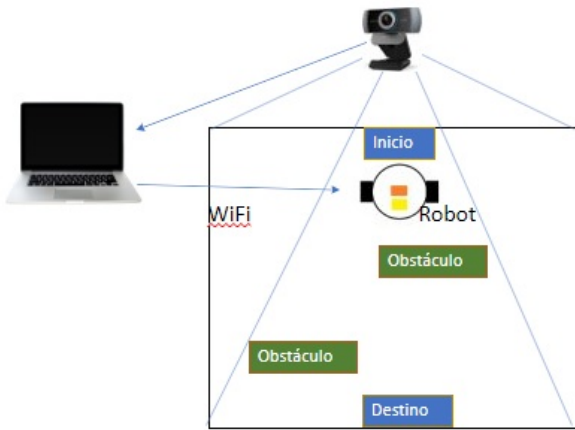


Figura 2. Diagrama General del Sistema



Figura 3. Sistema Real

II-B. Restricciones del Sistema

- El sistema funciona en interiores.

- El entorno está controlado; tanto en objetos presentes como la fuente de iluminación.
- La cámara fija percibe el punto de inicio, final, obstáculos, así como al mismo robot en todo momento.
- La ubicación del robot y los obstáculos es obtenida mediante visión artificial.
- El procesamiento y la generación de la ruta son realizados en un equipo de cómputo externo al robot.

III. PROCESAMIENTO DE IMAGEN

La visión es nuestro sentido más poderoso. Nos proporciona una enorme cantidad de información sobre el entorno y permite una interacción rica e inteligente en entornos dinámicos. Por lo tanto, no es sorprendente que se haya dedicado una gran cantidad de esfuerzo a proporcionar a las máquinas sensores que imiten las capacidades del sistema de visión humana. El primer paso en este proceso es la creación de dispositivos de detección que capturan la luz y la convierten en una imagen digital. El segundo paso es el procesamiento de la imagen digital para obtener información destacada como cálculo de profundidad, detección de movimiento, seguimiento de color, detección de características, reconocimiento de escenas, etc [1]. En Fig. 4 se puede observar un esquema general del procesamiento de imagen.

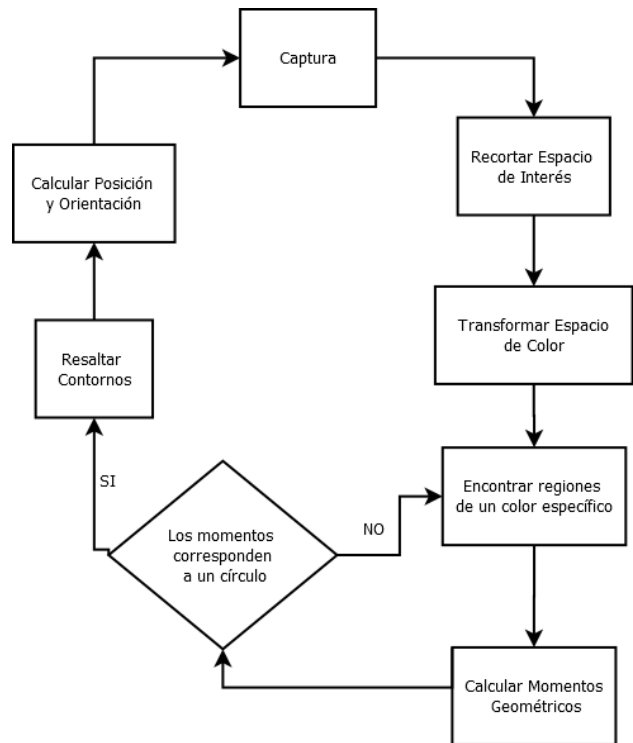


Figura 4. Diagrama Procesamiento de Imagen

III-A. Espacio de interés

Para construir el mapa capturamos la imagen en QT Creator con la ayuda de la librería OpenCV, se toma solo la región de

interés como se puede apreciar en Fig. 5 de la captura, utilizando `cv::Rect`. Obteniendo una imagen de tamaño 240x400 como puede observarse en “Fig. 6”.

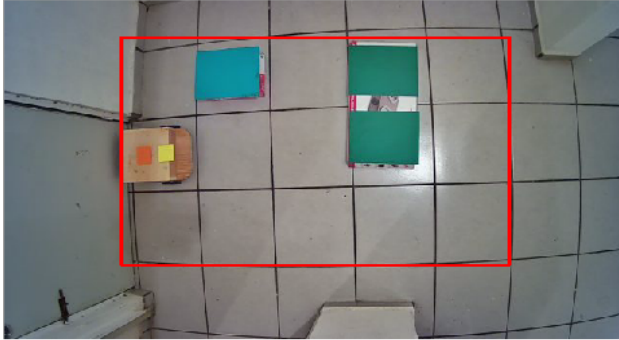


Figura 5. Captura de la cámara



Figura 6. Espacio de Interés

III-B. Segmentación por color

La imagen obtenida es procesada utilizando algunas funciones de la librería OpenCV para encontrar los obstáculos por su color y sus contornos. Primero se cambia el espacio de color de RGB a HSV utilizando `cv::cvtColor`. Dado que los colores en el espacio de color RGB se codifican utilizando los tres canales, es más difícil segmentar un objeto en la imagen según su color.

Sea MAX el valor máximo de los componentes (R, G, B), y MIN el valor mínimo de esos mismos valores, los componentes del espacio HSV se pueden calcular como:

$$H = \begin{cases} \text{No Definido,} & \text{si } MAX = MIN \\ 60^\circ \times \frac{G-B}{MAX-MIN} + 0^\circ, & \text{si } MAX = R \text{ y } G \geq B \\ 60^\circ \times \frac{G-B}{MAX-MIN} + 360^\circ, & \text{si } MAX = R \text{ y } G < B \\ 60^\circ \times \frac{B-R}{MAX-MIN} + 120^\circ, & \text{si } MAX = G \\ 60^\circ \times \frac{R-G}{MAX-MIN} + 240^\circ, & \text{si } MAX = B \end{cases} \quad (1)$$

$$S = \begin{cases} 0, & \text{si } MAX = 0 \\ 1 - \frac{MIN}{MAX}, & \text{en otro caso} \end{cases} \quad (2)$$

$$V = MAX \quad (3)$$

El espacio de color HSV (tono, saturación, valor) es un modelo para representar el espacio de color similar al modelo de color RGB. Dado que el canal de tono modela el tipo de color, es muy útil en tareas de procesamiento de imágenes que necesitan segmentar objetos en función de su color. El canal de valor describe el brillo o la intensidad del color. En Fig. 7 se puede apreciar el cilindro HSV.

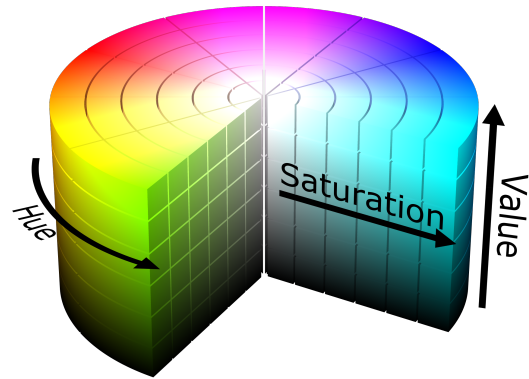


Figura 7. Cilindro HSV

Después se utiliza la función `cv::inRange` para la segmentación de colores, esta valida si los elementos de una matriz se encuentran entre los rangos específicos. La función verifica el rango de la siguiente manera:

Para cada elemento de una matriz de entrada de un solo canal:

$$dst(I) = lowerb(I)_0 \leq src(I)_0 \leq upperb(I)_0 \quad (4)$$

Donde src es la primera matriz de entrada, $lowerb$ es una matriz de límite inferior inclusiva o un escalar, $upperb$ es una matriz de límite superior inclusiva o un escalar y dst es una matriz de salida del mismo tamaño que src .

III-C. Encontrar contornos y centroides

Los contornos se pueden explicar simplemente como una curva que une todos los puntos continuos (a lo largo del borde), que tienen el mismo color o intensidad. Los contornos son una herramienta útil para el análisis de formas y la detección y reconocimiento de objetos. Para esto utilizamos `cv::findContours`. Para dibujar los contornos se utiliza la función `cv::drawContours`.

Para obtener el centroide se utiliza `cv::Moments`, esta función devuelve todos los momentos de un contorno. Los momentos geométricos son una medida particular que indica la dispersión de una nube de puntos, y matemáticamente se definen como:

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y) \quad (5)$$

donde x, y son las coordenadas de un píxel y la función $I(x, y)$ indica su intensidad. Estamos trabajando sobre una imagen

binaria, por tanto la función $I(x,y)$ sólo puede valer 0 ó 1. Podemos obtener los valores del centroide de la siguiente manera:

$$\frac{M_{10}}{M_{00}} = \frac{\sum_x \sum_y xI(x,y)}{\sum_x \sum_y I(x,y)} = \bar{x} \quad (6)$$

$$\frac{M_{01}}{M_{00}} = \frac{\sum_x \sum_y yI(x,y)}{\sum_x \sum_y I(x,y)} = \bar{y} \quad (7)$$

III-D. Posición y Orientación

La posición del robot se obtiene utilizando dos marcadores, uno naranja y otro amarillo, estos son localizados por su color, se encuentra su contorno y su centroide. Una vez localizados los marcadores se define la posición del robot en la coordenada del punto medio de la recta N .

La orientación del robot se basa en la pendiente que forman los centroides de los dos marcadores

En la Fig. 8 se observa la localización de los obstáculos y del robot.



Figura 8. Localización de los obstáculos y del robot

IV. CÁLCULO DE LA RUTA

Una vez encontrados los obstáculos se utiliza la función de OpenCV resize para cambiar el tamaño de la imagen y se obtiene una imagen de tamaño 3x5 píxeles. En la Fig. 9 se presenta el diagrama general para la generación del mapa.

Esta imagen Fig. 10 se utiliza como mapa M para encontrar una ruta desde un punto de inicio hasta un destino definidos.

Una vez obtenido un mapa y los puntos de inicio y final se procede a encontrar una ruta óptima utilizando el algoritmo A^* .

El nodo inicial y el nodo final se definen dentro del mapa, en el sistema. El ciclo del algoritmo comienza con la evaluación o expansión del primer nodo y determina los nodos vecinos. Estos nuevos nodos se consideran nodos "abiertos" mientras que los nodos analizados se consideran "cerrados". $f(x) = g(x) + h(x)$: donde $f(x)$ es el peso total o el costo total, mientras que $g(x)$ se denomina costo de ruta y $h(x)$ se denomina costo heurístico, la distancia más corta entre el nodo actual y el nodo final. El algoritmo elimina las rutas de alto costo y finalmente calcula la secuencia de nodos para la ruta más

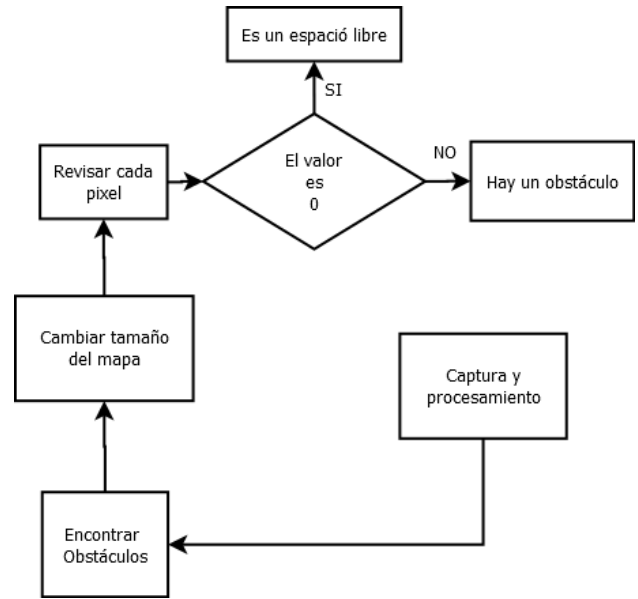


Figura 9. Generación del mapa



Figura 10. Mapa generado

corta. El pseudocódigo del algoritmo puede observarse en el Algoritmo 1.

```

ABIERTOS(M) ← [INICIAL]
CERRADOS(M) ← []
f'(INICIAL) ← h'(INICIAL)
mientras SOLUCION_ENCONTRADA
≠ TRUE hacer
  si ABIERTOS = [] entonces
    | FALLO
  en otro caso
    extraer MEJORNODO de ABIERTOS con f'
    mínima
    mover MEJORNODO de ABIERTOS a
    CERRADOS
    si MEJORNODO contiene estado_objetivo
    entonces
      | SOLUCION_ENCONTRADA ← TRUE
    en otro caso
      generar SUCESORES de MEJORNODO
      para cada SUCESOR hacer
        TRATAR_SUCESOR
      fin
    fin
  fin

```

Algoritmo 1: Pseudocódigo del algoritmo de búsqueda A*

V. SEGUIMIENTO DE LA TRAYECTORIA

V-A. Control

Una vez encontrada la ruta es necesario implementar dos algoritmos de control, uno para controlar la orientación y otro para la posición.

Para controlar la posición del robot se utiliza un control Proporcional, donde el error $e_p(t)$ se define como la diferencia lineal en pixeles entre la posición actual del robot y la posición del siguiente punto en la trayectoria P_s [8]. En la Fig. 11 se observa el diagrama del error de posición.

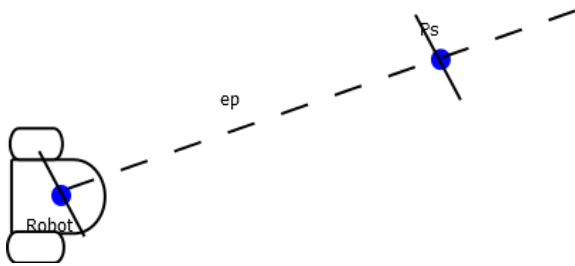


Figura 11. Diagrama error de posición

La relación entre la salida del controlador $u_p(t)$ y la señal de error $e_p(t)$ es:

$$u_p(t) = k_{pp}e_p(t) \tag{8}$$

donde k_{pp} se considera la ganancia proporcional [8].

Para controlar la orientación del robot se utiliza un control Proporcional-Integral donde el error $e_o(t)$ se define como la diferencia entre la pendiente del robot N y la pendiente de la línea que se forma entre el centro del robot y el siguiente punto en la trayectoria (L). En la Fig. 12 se observa el diagrama del error de orientación.

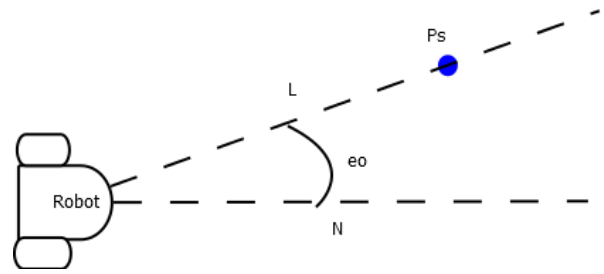


Figura 12. Diagrama error de orientación

La acción de control proporcional-integral es:

$$u_o(t) = k_{po}e_o(t) + k_i \int_0^t e_o(t) \tag{9}$$

donde k_i se considera la ganancia integral.

V-B. Envío y recepción de datos

Las señales de control de posición y orientación se normalizan entre -1 y 1 y son enviadas mediante una conexión TCP utilizando la clase QTcpSocket.

Las señales de control son recibidas por un módulo ESP8266, el cual se comunica a través del puerto serial a un Arduino Mega.

El código programado en Arduino recibe las señales y las interpreta para generar valores PWM que se aplican a los puentes H L298 que a su vez controlan los dos motores del robot. En la Fig. 13 se observa el diagrama envío y recepción de datos.

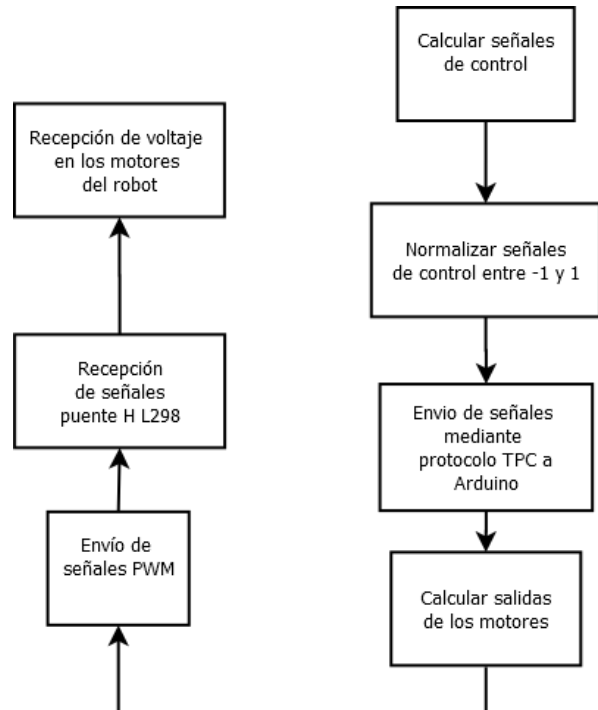


Figura 13. Diagrama envío y recepción de datos

VI. RESULTADOS

En la Fig. 14 se muestra el proceso general de la generación de la ruta en 14.(a) se observa la captura de la cámara desde un plano superior, en 14.(b) se muestra el mapa construido en el cual se distinguen espacios libres y espacios con obstáculos, en 14.(c) se muestra la ruta obtenida utilizando el algoritmo A*. Desde la captura de la imagen hasta la generación de la ruta hay un tiempo de ejecución promedio de 33ms.

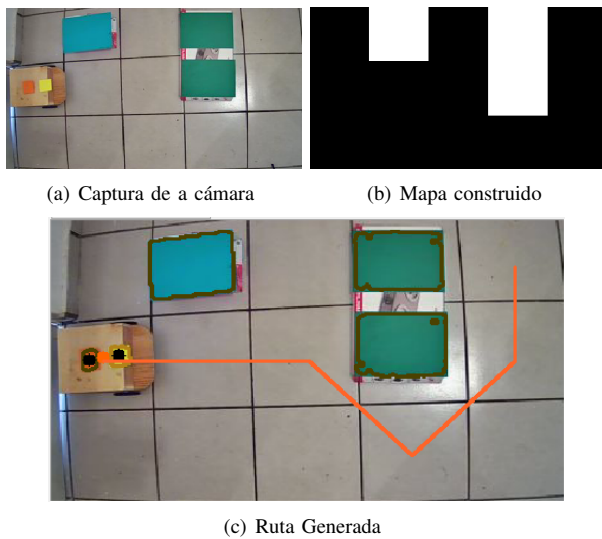


Figura 14. Proceso de generación de la ruta

Al mover los obstáculos el algoritmo encuentra una ruta distinta para llegar al destino como puede observarse en Fig. 15.

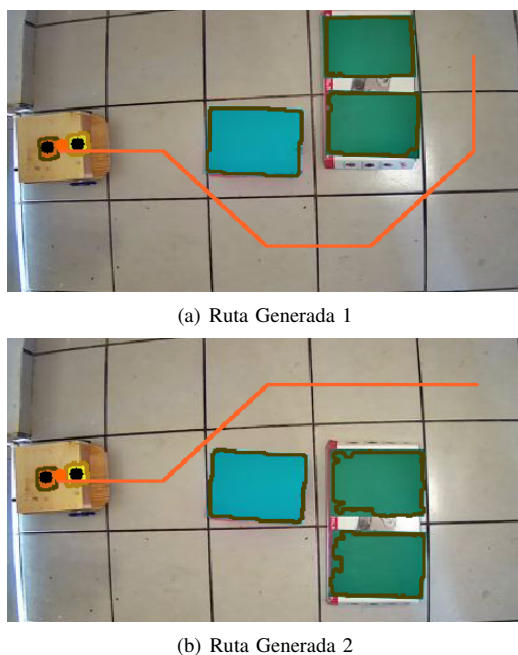


Figura 15. Rutas Generadas cambiando los obstáculos

VII. CONCLUSIONES Y TRABAJO FUTURO

Este artículo propone una plataforma de bajo costo para evaluar sistemas de navegación. El sistema en general y los algoritmos son fáciles de replicar y se comprobó su funcionamiento, se utilizaron plataformas de código abierto así como componentes asequibles minimizando el costo total del sistema.

El sistema propuesto anteriormente funciona considerando obstáculos estáticos; como trabajo futuro puede ampliarse para trabajar en un entorno dinámico mejorando los algoritmos de navegación, o utilizando diferentes algoritmos. La plataforma puede funcionar mejor con otros algoritmos de control o sintonizando el controlador. El uso de un sistema estéreo podría permitir discriminar objetos sobre los que pueda pasar el robot, e ignorarlos como obstáculos.

REFERENCIAS

- [1] R. Siegwart, I. Nourbakhsh and D. Scaramuzza, Introduction to autonomous mobile robots. Cambridge, Mass.: MIT Press, 2011.
- [2] A. Ollero Baturone, Robotica, manipuladores y robots m'oviles. Barcelona: Marcombo, 2001.
- [3] F. Giamarchi, Robots moviles. Madrid: Paraninfo, 2001.
- [4] Flinn, J., Ortiz, H.S., Yuan, S.: A secure routing scheme for networks with unknown or dynamic topology using A-star algorithm. In: International Conference Security and Management (2016)
- [5] Jiang, B., Wang, Y., Zhao, Z.: Path planning for terrestrial platform based on A-star algorithm. In: 3rd International Conference on Mechatronics, Robotics and Automation. 409– 414 (2015)
- [6] S. Sheth A. Ajmera, A. Sharm, S. Patel, Shivang, C. Kathrecha, "Design and Development of Intelligent AGV Using Computer Vision and Artificial Intelligence,"Advances in Intelligent Systems and Computing. 337-349 (2018)
- [7] Parikh, P.A., Joshi, K.D., Sheth, S.: Color guided vehicle—an intelligent material handling mechatronic system. In: 1st International and 16th National Conference on Machines and Mechanisms. 628–635 (2013)
- [8] K. Ogata, Ingenieria de control moderna. Pearson Educacion de Mexico, S.A. de C.V., 2011.
- [9] R. Szeliski., Computer Vision: Algorithms and Applications. Springer. (2010).
- [10] R. Zurita, A. Mora, C. Alvarez, M. Lechner: Sistema de localización para robots móviles de bajo costo utilizando marcas de referencia artificiales en ambientes de interiores. XXV Congreso Argentino de Ciencias de la Computación (CACIC). 1081-1090 (2019).
- [11] D. Leon, Diseno de una estrategia para la planeacion de rutas de navegacion autonoma de un robot movil en entornos interiores usando un algoritmo de aprendizaje automatico. Tesis, UNC, Medelln, Colombia (2020).

Detección de automóviles en un estacionamiento haciendo uso de las redes neuronales pulsantes

Víctor Romero Bautista

Facultad de Ciencias de la Electrónica (FCE)
Benemérita Universidad Autónoma de Puebla (BUAP)
Puebla, México
victor.romerobau@alumno.buap.mx

Aldrin Barreto Flores

Facultad de Ciencias de la Electrónica (FCE)
Benemérita Universidad Autónoma de Puebla (BUAP)
Puebla, México
aldrin.barreto@correo.buap.mx

Salvador E. Ayala Raggi

Facultad de Ciencias de la Electrónica (FCE)
Benemérita Universidad Autónoma de Puebla (BUAP)
Puebla, México
salvador.raggi@correo.buap.mx

José F. Portillo Robledo

Facultad de Ciencias de la Electrónica (FCE)
Benemérita Universidad Autónoma de Puebla (BUAP)
Puebla, México
francisco.portillo@correo.buap.mx

Verónica E. Bautista López

Facultad de Ciencias de la Computación (FCC)
Benemérita Universidad Autónoma de Puebla (BUAP)
Puebla, México
veronica.bautistalo@correo.buap.mx

Resumen—Uno de los retos de los sistemas para detección de espacios disponibles en estacionamientos basados en visión por computadora es generar una adecuada segmentación de los automóviles, con el fin de obtener un alto porcentaje de efectividad, algunos trabajos que se han propuesto utilizan técnicas como el mapeo del estacionamiento de interés, detección de bordes y esquinas, o bien usando redes neuronales convolucionales, donde algunos de los problemas que se enfrentan en estos sistemas suelen ser los cambios intensos de iluminación, las diferentes tonalidades de color que pueden presentar los automóviles, oclusión y efectos de sombra. En este trabajo se propone un método para llevar a cabo la detección de automóviles en un estacionamiento basado en el uso de la red ICM, tanto para el proceso de segmentación como para la extracción de características utilizando la firma que genera esta red y el clasificador MLP para realizar la detección de los automóviles. Se llevaron a cabo pruebas del método propuesto con un conjunto de 123 imágenes de un estacionamiento, donde se presentó un porcentaje de efectividad del 95 %.

Palabras clave—redes neuronales, características, segmentación, reconocimiento, detección.

I. INTRODUCCIÓN

Los sistemas de asistencia a estacionamientos proporcionan información acerca de la disponibilidad de un espacio de estacionamiento, estos sistemas a su vez, contribuyen a reducir el tráfico en los mismos, permiten mantener un control adecuado de los automóviles dentro de ellos, entre otras ventajas. A lo largo de los años se han propuesto métodos basados en visión por computadora para llevar a cabo sistemas de asistencia a estacionamiento, y dentro de éstos, uno de los temas de interés, es la detección de lugares disponibles en un estacionamiento, en donde algunos de los principales inconvenientes de estos

sistemas se exponen en [1], siendo algunos de éstos, la oclusión, sombras, tonalidades de color que presentan los automóviles y los cambios intensos de luminosidad. Para enfrentar estos retos se han propuesto diferentes métodos, tales como el reconocimiento de bordes y esquinas presentados en [2] y [3], implementando un mapeo del estacionamiento como el que se presenta en [1] y [4], mediante la resta del fondo del estacionamiento, o bien haciendo uso de redes neuronales convolucionales profundas como en [5] y [6].

Los métodos basados en un mapeo del estacionamiento logran alcanzar una alta efectividad en la detección, y debido a que solo se enfocan a regiones predeterminadas, presentan baja susceptibilidad a los efectos de sombra, pero tienen la restricción de que la cámara de captura debe permanecer la misma posición y de ser cambiada el sistema pierde de manera drástica su efectividad; los sistemas que aplican la resta del fondo del estacionamiento presentan este mismo problema, pero además también se enfrentan a elegir los valores correctos de intensidad del fondo para ejecutar la resta, ya que éstos suelen variar a lo largo del día, estos sistemas presentan baja efectividad debido a los diversos efectos de sombra que suelen provocar los automóviles así como a la variación de iluminación que ocurre a lo largo del día; por otro lado, los métodos basados en la detección de bordes y esquinas están enfocados a la detección de las marcas que corresponden a los delimitadores de los espacios de estacionamiento, éstos presentan una buena respuesta ante el problema de las sombras pero están sujetos a que el estacionamiento donde serán implementados mantenga de manera adecuada la uniformidad de las marcas de los espacios de estacionamiento; una de las técnicas

recientes es el uso de las redes neuronales convolucionales, las cuales han presentado una buena respuesta ante las distorsiones ocasionadas por las sombras, sin embargo, su efectividad se ve afectada en imágenes de baja iluminación, esto además de requerir de un tiempo de entrenamiento elevado.

Una técnica poco explorada para la detección de espacios disponibles en estacionamientos, es el uso de redes neuronales pulsantes (PCNN); este tipo de redes presenta características que pueden ser favorables para el reconocimiento de objetos, ya que se pueden emplear para procesos de segmentación, extracción de características y detección de bordes, [7], [8], [9].

En este trabajo se presenta un algoritmo para la detección de lugares disponibles en un estacionamiento basado en el uso de las redes neuronales pulsantes (PCNN) en su forma simplificada, el modelo ICM, para los procesos de segmentación y extracción de características, en donde se proponen valores de sus parámetros para estos modos de operación; en la etapa de clasificación se empleó la red neuronal Perceptrón Multicapa (MLP), en donde se realizaron pruebas de validación cruzada con diferentes arquitecturas de la red, con el objetivo de obtener la arquitectura que arroje mejores resultados para su implementación. Se realizaron pruebas con 123 imágenes de un estacionamiento, de acuerdo con los resultados obtenidos, el método propuesto presentó un porcentaje de efectividad del 95 % en la detección de automóviles en imágenes de un estacionamiento. Sin embargo, el algoritmo propuesto presenta una disminución de su efectividad en imágenes con alta iluminación, la cual provoca efectos de sombra en los automóviles.

II. MARCO TEÓRICO

A continuación se presentan los conceptos teóricos sobre los que se sustenta este trabajo, donde se presenta primeramente el modelo de redes neuronales pulsantes, en seguida el clasificador MLP y posteriormente el modelo de color HSV.

A. Las redes neuronales pulsantes

Las redes neuronales pulsantes (PCNN), del inglés *Pulse-Coupled Neural Networks*, están basadas en el modelo cortical propuesto por Eckhorn, el cual a su vez está inspirado en la sincronización de los pulsos dentro de la corteza visual de los mamíferos y han sido aplicadas ampliamente en el procesamiento de imágenes y reconocimiento de patrones [10].

Si el estímulo en la PCNN es una imagen, entonces la red estará compuesta por tantas neuronas como píxeles se tengan en dicha imagen, y la red emitirá una serie de pulsos de imágenes Y , que representan características fundamentales de la imagen original, tales como bordes y textura, como se muestra en la fig. 1. Una neurona pulso-acoplada recibe información de sus neuronas vecinas representadas por Y_{kl} , a través las sinapsis M , y es activada de forma síncrona en ciertas regiones, es por esto que la red PCNN puede ser aplicada en la segmentación y codificado de imágenes; otra importante característica de esta red es que los pulsos de imágenes pueden ser caracterizados en una firma única para la identificación de imágenes. Algunas ventajas a destacar

sobre otros métodos, es qué por la naturaleza de esta red, es posible realizar el procesamiento en paralelo mientras que en los algoritmos tradicionales se suele procesar la imagen píxel a píxel o por regiones, donde se hace notorio el tiempo de procesado a medida en que el tamaño de la imagen aumenta, otro aspecto importante es que la imagen conserva propiedades de escalado y rotación [11].

B. El modelo ICM

Debido al potencial que presentan las redes PCNN, se han desarrollado variantes con base en el modelo original ya sea simplificando su estructura o mejorando su desempeño. Uno de estos modelos es el Modelo de Intersección Cortical o ICM del inglés *Intersection Cortical Model* [12], el cual busca reducir el costo computacional mediante la eliminación de algunos parámetros del modelo PCNN original; siendo un caso especial en donde cuatro de los siete parámetros del modelo original son establecidos en valor cero, por lo tanto, el modelo ICM solo requiere de 3 parámetros los cuales son, los factores de decaimiento f para el canal *Feeding* y g para el umbral dinámico θ , y el potenciador h . Una neurona ICM (fig.2) está compuesta por tres compartimientos, los cuales son: alimentación, potencial interno y generador de pulsos. En el compartimiento de alimentación se obtiene la información de las neuronas vecinas en un tiempo anterior $Y_{kl}[n - 1]$, dicha información es acotada por la matriz de pesos sinápticos M ; el potencial interno es representado por el canal receptivo *Feeding* (F_{ij}), en donde se genera la sumatoria entre la información de las neuronas vecinas $Y_{kl}[n - 1]$, el valor de intensidad S_{ij} correspondiente a la imagen original y el valor del potencial interno generado en un tiempo anterior $F_{ij}[n - 1]$ multiplicado por un factor de decaimiento f ; y en el compartimiento del generador de pulsos ocurren dos procesos, el primero se encarga de generar el valor del umbral dinámico definido por θ_{ij} , y el segundo proceso obtiene la salida de la neurona Y_{ij} , comparando el valor de θ_{ij} con F_{ij} ; en donde si F_{ij} es mayor, se produce la activación de la neurona, de lo contrario ésta permanece en valor 0. El funcionamiento de la red ICM se describe mediante las ecuaciones (1), (2) y (3), calculadas en ese mismo orden.

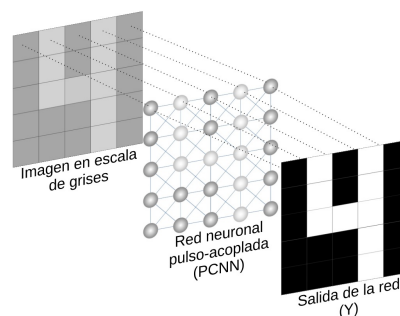


Fig. 1. Representación de la red PCNN.

$$F_{ij}[n] = fF_{ij}[n - 1] + \sum_{kl} M_{ijkl}Y_{kl}[n - 1] + S_{ij} \quad (1)$$

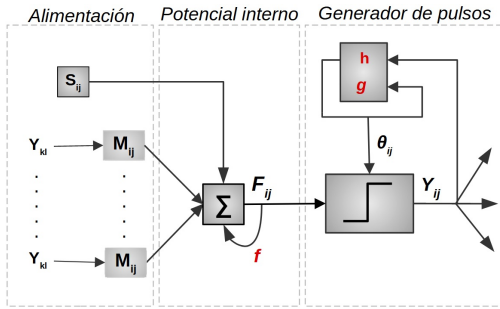


Fig. 2. Representación de una neurona ICM.

$$\theta_{ij}[n] = g\theta_{ij}[n-1] + hY_{ij}[n-1] \quad (2)$$

$$Y[n] = \begin{cases} 1 & \text{si } F_{ij}[n] > \theta_{ij}[n] \\ 0 & \text{en caso contrario} \end{cases} \quad (3)$$

C. El perceptrón multicapa

La red perceptrón multicapa (MLP), es un clasificador basado en aprendizaje supervisado, el cual a través de una fase de entrenamiento donde se dota a la red de una serie de información, es capaz de aprender, para posteriormente ser usado como un clasificador de nuevos datos, esta red es conocida por su amplio rango de aplicación en multitud de problemas de diferentes áreas del conocimiento y es considerada una de las arquitecturas más versátiles gracias a su aplicabilidad, siendo algunas de sus principales aplicaciones el reconocimiento de patrones o bien su uso como un aproximador universal de funciones [13]. La MLP consiste en un arreglo de varias neuronas de tipo perceptrón simple, con la diferencia de que la función de activación es de tipo sigmooidal o tangente hiperbólica. En la fig. 3, se muestra la representación de una red perceptrón multicapa, la cual cuenta con una capa de entrada C_{en} , dos capas ocultas CO_1 y CO_2 , y una capa de salida C_s . La cantidad de neuronas en las capas ocultas puede establecerse desde dos, hasta n neuronas, mientras que en la capa de salida, el número de neuronas dependerá del problema a resolver, donde si se trata de un clasificador binario, requerirá de una sola neurona, o si se trata de un clasificador de múltiples clases, entonces se necesitarán dos o más neuronas.

La matriz de pesos sinápticos w_1 conecta las neuronas de C_{en} con las neuronas de CO_1 , por consiguiente w_2 conecta las neuronas de la CO_1 con las de CO_2 , y w_3 conecta CO_2 con C_s . El algoritmo usualmente utilizado para el proceso de aprendizaje de la red MLP, es el *backpropagation* o retropropagación, el cual está basado en el método del descenso del gradiente [14], y que se encarga de ajustar el valor de los pesos sinápticos de tal manera que, por cada ajuste se logre disminuir el error obtenido por la red, esto se hace de forma iterativa hasta alcanzar un límite de error mínimo o bien una cantidad de épocas máximo.

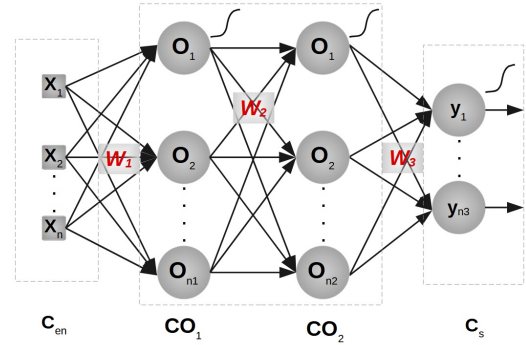


Fig. 3. Ilustración de la red Perceptrón multicapa.

D. El espacio de color HSV

Este modelo representa de mejor manera la forma en como los humanos percibimos el color en comparación con el modelo RGB, y lo hace a través de tres componentes, Tonalidad (Hue), Saturación (Saturation) y Valor (Value) [15]; estos componentes a su vez representan un sistema coordenado cilíndrico como se muestra en la fig. 4; donde la tonalidad indica el color, y varía desde 0 a 360 grados, la saturación indica que tan intenso es el color y el valor indica el brillo. Este modelo de color se puede obtener a partir de una imagen RGB mediante las ecuaciones (4), (5) y (6) [16].

$$H = \arccos \frac{\frac{1}{2} * 2 * R - G - B}{\sqrt{(R - G)^2 + (R - B) * (G - B)}} \quad (4)$$

$$S = 1 - 3 \frac{\min(R, G, B)}{R + G + B} \quad (5)$$

$$V = \frac{1}{3}(R + G + B) \quad (6)$$

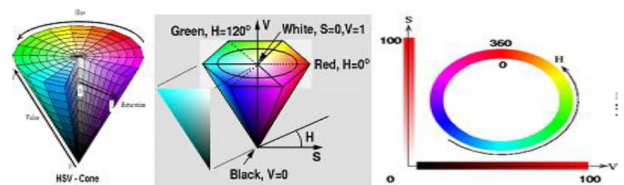


Fig. 4. Representación del espacio de color HSV [16].

III. RECONOCIMIENTO DE OBJETOS EN UNA IMAGEN

Como primer paso para llevar a cabo el reconocimiento de los objetos en una imagen se accedió a un conjunto de 123 imágenes con diferentes capturas de un estacionamiento, donde cada imagen fue tomada en diferente tiempo. Se realizó una selección variada de 36 imágenes del conjunto, las cuales se denominaron como imágenes de entrenamiento, para ser utilizadas en el proceso de extracción de características y entrenamiento del clasificador MLP; y las 86 imágenes

restantes fueron empleadas para determinar la efectividad del sistema. En la fig. 5, se muestra el proceso llevado a cabo para el reconocimiento de los objetos, desde la obtención de una imagen, representado por el inciso (a); posteriormente se lleva a cabo el pre-procesamiento de la misma en los incisos (b) y (c); continuando con el proceso de segmentación en el inciso (d); en el inciso (e) se aplica el etiquetado de la imagen segmentada; mientras que en los incisos (f) y (g) se realiza la extracción de características; y por último la clasificación de éstas se lleva a cabo en el inciso (h). A continuación, se explicarán los puntos de mayor importancia del método propuesto.

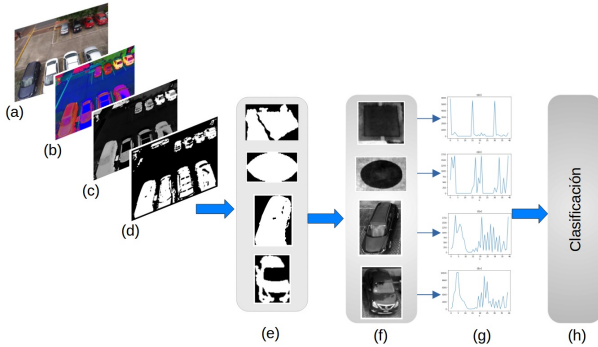


Fig. 5. Diagrama del método propuesto para el reconocimiento de objetos en una imagen; (a) imagen original en rgb (I_{rgb}); (b) imagen original en espacio de color HSV (I_{hsv}); (c) espectro Hue (I_h); (d) imagen binaria, resultado del proceso de segmentación por ICM (I_b); (e) extracción de los objetos de la imagen binaria mediante el etiquetado de la misma; (f) representación de los objetos en escala de grises, (g) firma de los objetos $G[n]$; (h) etapa de clasificación de las características.

A. Pre-procesamiento

En esta etapa se realiza la conversión de la imagen original en espacio de color RGB a HSV mediante la relación presentada en las ecs. (4), (5) y (6); posteriormente, de la imagen en espacio de color HSV (I_{hsv}), se extrae el espectro Hue (I_h). Para eliminar pequeñas partículas en la imagen I_h , se aplica la operación morfológica de apertura para imágenes en escala de grises, esta operación implementa primeramente un filtro de erosión seguida del filtro de dilatación; en las ecs. (7) y (8) se presenta la relación de estos operadores respectivamente, en donde I representa una imagen en escala de grises; w corresponde al tamaño del filtro utilizado para la operación dado en filas y columnas; mientras que i, j representa la posición de un píxel en la imagen I ; y en la ec. (9) se indica la operación de apertura para una imagen en escala de grises.

$$E(I) = \min_{(s,t) \in w} \{I(i-s, j-t)\} \quad (7)$$

$$D(I) = \max_{(s,t) \in w} \{I(i-s, j-t)\} \quad (8)$$

$$A(I) = D(E(I)) \quad (9)$$

B. Segmentación

El proceso de segmentación es una etapa crucial para el reconocimiento de los objetos en una imagen, ya que en esta etapa se intenta separar el fondo, de los objetos de interés en la imagen. Existen diferentes técnicas de segmentación, una de las más comúnmente utilizadas es el método de Otsu [17], ya que presenta una susceptibilidad baja al cambio de iluminación, sin embargo, de acuerdo a una serie de pruebas que se llevaron a cabo para la segmentación con el conjunto imágenes del estacionamiento, este algoritmo de segmentación presentó un bajo rendimiento. Debido a lo anterior se decidió hacer uso de la red ICM para la segmentación de las imágenes con el fin de aprovechar la información de los diferentes pulsos que la red genera durante su funcionamiento.

Se utilizó la entropía para medir la información de los pulsos que arroja la red ICM, siendo que si una imagen binaria tiene una medida de entropía elevada, esto quiere decir que presenta una buena segmentación. Para calcular la entropía de una imagen binaria se hace uso de la ec. (10), donde p_0 corresponde a la probabilidad de obtener el valor 0, y p_1 para el valor 1.

$$H[n] = -p_0[n] \log_2 p_0[n] - p_1[n] \log_2 p_1[n] \quad (10)$$

C. Extracción de características

El proceso de extracción de características se encarga de cuantificar un objeto dentro de un vector, estas deben ser repetibles y robustas a cambios como el ruido, escalado o rotación. Para esto, se hace uso de la firma que genera la red ICM; en [18], [19], y [20], se llevan a cabo trabajos de detección de objetos basados en la firma producida por la red PCNN, en donde se logran obtener buenos resultados, esto debido a que la firma generada es invariable al escalado y rotación.

La firma que produce la red ICM contabiliza en el número de neuronas $Y_{i,j}$ que han sido activadas en una iteración $[n]$ y se representa por $G[n]$ dada por la ecuación 11. Un aspecto importante a tomar en cuenta al momento de capturar la firma de un objeto, es la cantidad de pulsos o iteraciones que se deben producir.

$$G[n] = \sum_{ij} Y_{ij}[n] \quad (11)$$

D. Clasificación

Para este proceso se hace uso de la red MLP, y como se muestra en la fig. 5, el clasificador recibe la firma generada por la red ICM de acuerdo a los objetos extraídos en una imagen. Antes de hacer uso del clasificador, éste debe ser entrenado para ajustar sus pesos sinápticos, los cuales se encargan de enlazar la información entre las neuronas de cada capa, para que dada una muestra, se obtenga un valor de salida igual o aproximado al deseado. El algoritmo de entrenamiento que se utilizó fue el *backpropagation* o retropropagación. Se utilizaron 36 imágenes de un estacionamiento para extraer las características de los objetos dentro de las mismas, donde se

recabaron 340 muestras, de las cuales 170 corresponden a características de automóviles y las restantes pertenecen a objetos que no lo son. Cada muestra consta de 40 características que corresponden a la firma del objeto $G[n]$.

IV. RESULTADOS EXPERIMENTALES

A continuación, se presenta la configuración de los filtros de dilatación y erosión utilizados para la operación de apertura, los parámetros de la red ICM para segmentación y obtención de la firma, la arquitectura de la red MLP que fue utilizada y los resultados experimentales obtenidos de la implementación del método propuesto para la detección de los automóviles.

A. Estructuras de referencia para la operación de apertura

La operación de apertura para imágenes en escala de grises se lleva a cabo mediante la aplicación del filtro de erosión, y a la imagen resultante se le aplica el filtro de dilatación. Para la aplicación de los filtros mencionados se requiere definir la estructura de referencia para cada uno, la cual define la influencia que tienen los píxeles vecinos sobre un píxel central. Las estructuras de referencia utilizadas para la aplicación de la operación de apertura que se proponen en este trabajo se presentan en las ecs. (12) y (13), para aplicar los filtros de erosión y dilatación respectivamente.

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad (12)$$

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (13)$$

B. Parámetros para segmentación

Una de las principales tareas a la hora de implementar la red ICM es la elección adecuada de sus parámetros, los cuales determinan el tipo de información que la red puede emitir en su salida; de acuerdo con esto, se llevó a cabo la sintonización de estos parámetros bajo una serie de pruebas, en donde se dedujeron los valores para la segmentación, los cuales se presentan en la tabla I. Así mismo, la matriz de pesos sinápticos M que se utilizó, se indica en la ec. (14). La cantidad de iteraciones para segmentación se estableció debido a que se observó que a partir de la décima iteración la red comenzaba a repetir el patrón de los pulsos de salida $Y[n]$. Para obtener la mejor segmentación, se realizó la integración de tres pulsos, los cuales son: el primer pulso, con el cual se obtiene la información de automóviles en color blanco, y los dos pulsos de mayor entropía, de los cuales se obtiene la información de las demás tonalidades de color.

C. Parámetros para $G[n]$

Al igual que para el proceso de segmentación, se establecieron los parámetros de la red ICM para obtener la firma del objeto $G[n]$ en base a pruebas, donde se buscó que el patrón de la firma generado por los diferentes automóviles resultara similar, mientras que para los objetos que no corresponden a automóviles presentarán un patrón distinto. En la tabla I se presentan los parámetros de la red que se obtuvieron en este proceso, así mismo, la cantidad de pulsos se estableció de acuerdo a que se observó, que a partir del pulso 56 se repite el patrón de la firma, además de esto se observó que los primeros 15 pulsos siempre se conservan en valor 0, por lo cual se decidió eliminar estos pulsos, y quedarse únicamente con los 40 resultantes, los cuales representan a los objetos.

Tabla I
PARÁMETROS DE LA RED ICM.

Modo de uso	f	g	h	Iteraciones
Segmentación	0.19	0.6	19	10
Firma	0.01	0.8	20	55

$$M = \begin{pmatrix} 0,1 & 0,5 & 0,1 \\ 0,5 & 0 & 0,5 \\ 0,1 & 0,5 & 0,1 \end{pmatrix} \quad (14)$$

D. Configuración de la red MLP

El clasificador MLP puede ser configurado tanto en el número de capas ocultas, como en la cantidad de neuronas en cada una de éstas; para elegir una adecuada arquitectura a implementar, se decidió llevar a cabo pruebas de validación cruzada con diferentes arquitecturas, y elegir la que obtenga una mejor efectividad. Para esto, se utilizaron un total de 340 muestras, de las cuales 170 corresponden a muestras de firmas generadas por automóviles y 170 de objetos que no lo son, éstos se presentan debido a que no fueron eliminados durante el proceso de segmentación, los cuales pueden ser partes del piso y techo del estacionamiento o bien señalamientos que se encuentran dentro del mismo; para la fase de aprendizaje se estableció el número máximo de épocas a 5500, y una tasa de aprendizaje de 0.00025.

Para el proceso de validación cruzada se implementaron 10 bloques de 34 muestras, y se realizaron pruebas con una arquitectura de red MLP de dos capas ocultas y diferentes configuraciones de neuronas en cada una de estas capas respetando una arquitectura piramidal, se comenzó desde 3 hasta 60 neuronas en la primera capa oculta, y de 2 a 35 neuronas en la segunda capa oculta. Los resultados obtenidos se presentan en la fig. 6, donde se puede observar que la arquitectura que incorpora 30 neuronas en la primera capa oculta y 20 en la segunda, obtuvo ligeramente una mayor efectividad, la cual fue del 96.4%, y por lo tanto fue seleccionada para la implementación en la etapa de clasificación. En la fig. 7 se presenta la gráfica del decremento del error obtenida del proceso de aprendizaje de la red MLP con la arquitectura

de mayor efectividad, así mismo, en la fig. 8 se muestra la representación de ésta.

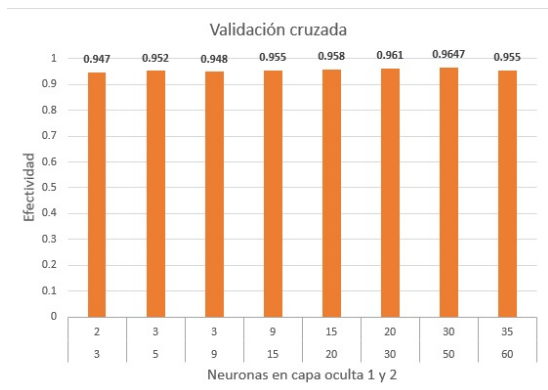


Fig. 6. Resultados de validación cruzada obtenidos con diferentes arquitecturas de red MLP, el eje x muestra la configuración de neuronas en cada capa oculta de la red, la primer fila muestra la cantidad de neuronas en la capa oculta 2 (CO_2), mientras que la segunda fila presenta las neuronas en la capa oculta 1 (CO_1).

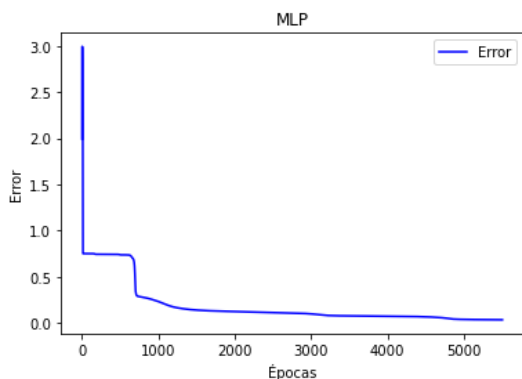


Fig. 7. Gráfica del error obtenido en la fase de entrenamiento de la red MLP.

E. Conjunto de muestras

Las 123 imágenes que se utilizaron para este trabajo son originalmente de tamaño 3264×2448 píxeles, y fueron redimensionadas a tamaño VGA para su procesamiento, en la fig. 9 se presentan las diferentes tomas con las que se cuentan en el conjunto de imágenes. Cada toma presenta al menos una sucesión de 12 capturas en diferente tiempo, de las cuales se tomaron 36 imágenes para el proceso de aprendizaje y validación de la red MLP, y las 86 restantes se utilizaron para probar la efectividad del método propuesto.

En la fig. 10 se muestran algunos de los objetos extraídos de las imágenes del estacionamiento después del pre-procesamiento y segmentación, en el inciso (a), (b) y (c), se muestran algunos automóviles y en los incisos (d) y (e) se presentan objetos que no pudieron ser eliminados en proceso de segmentación, los cuales pueden representar partes del estacionamiento como pueden ser el techo, el piso o señalamientos. Así mismo, en la fig. 11 se muestra el patrón

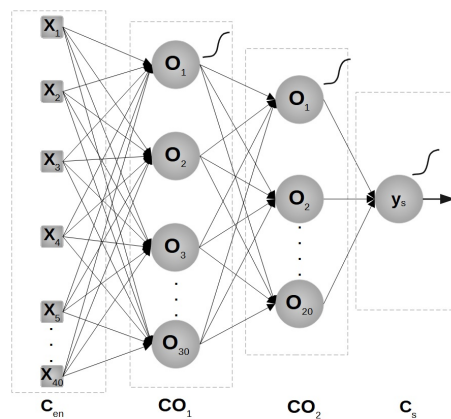


Fig. 8. Arquitectura de red MLP utilizada para el proceso de clasificación.

de las firmas $G[n]$ obtenidas por los autos y los objetos que no representan automóviles; en el inciso (a) se muestra la firma que genera un auto y en el inciso (b) se muestra el conjunto de firmas obtenidas de los automóviles a partir de la extracción de características, mientras que en el inciso (c) se presenta el patrón de firma producido por objetos que no son automóviles y en el inciso (d) se presenta el conjunto de firmas recabadas por este tipo de objetos, las cuales fueron obtenidas a partir del proceso de extracción de características.

F. Resultados experimentales

A continuación, se presentan los resultados obtenidos durante la implementación del método propuesto, que está representado en el algoritmo 1, el cual fue evaluado con las 86 imágenes restantes del conjunto. En la fig. 12 se muestran algunas capturas de los resultados obtenidos, en donde se encierran en recuadros de color rojo los objetos detectados como automóviles.

Las métricas utilizadas para evaluar el algoritmo fueron la efectividad y precisión, en las ecs. (15) y (16) se presenta la relación matemática para cada una respectivamente, donde VP corresponde a los verdaderos positivos, FP indica los falsos positivos y M es la cantidad de muestras sobre las cuales se efectúa la evaluación [21]; además se evaluó el tiempo de ejecución del algoritmo, el cual fue implementado en lenguaje *Python* y se ejecutó sobre un procesador *Intel Core i5 8400T* con velocidad de CPU a 1.70GHz y 16 GB de memoria RAM. En la tabla II se muestran los resultados de la evaluación para las 86 imágenes, en donde se tienen un total de 504 automóviles en total, de los cuales se logró detectar 483; debido a que en la mayoría de las imágenes de prueba se obtuvo un reconocimiento de todos los automóviles. La efectividad obtenida fue de 95.83 %, sin embargo, debido a la presencia de falsos positivos, la precisión obtenida fue del 92 %, y el promedio del tiempo de ejecución del algoritmo por imagen fue de 3.16 segundos.

$$\text{Efectividad} = \frac{VP + FP}{M} \quad (15)$$

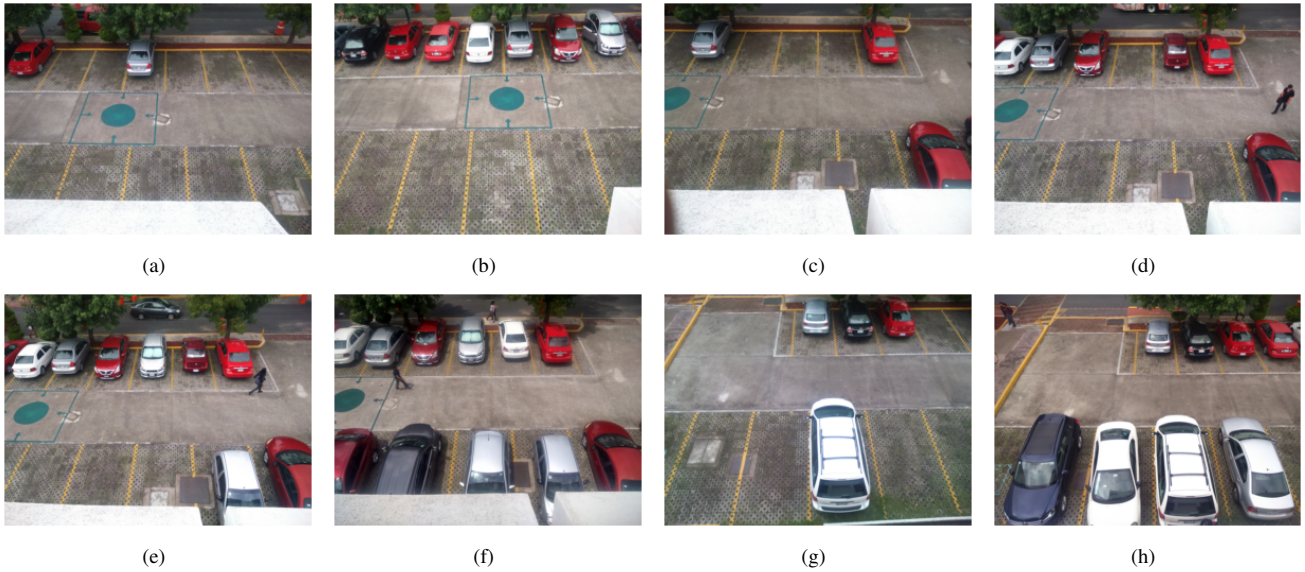


Fig. 9. Capturas pertenecientes al conjunto de 123 imágenes de un estacionamiento.

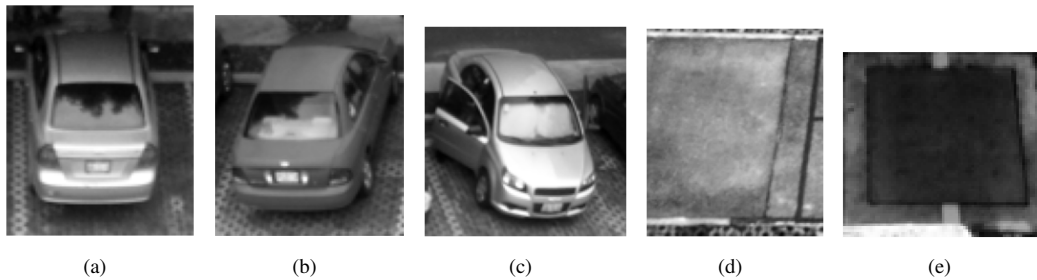


Fig. 10. Objetos extraídos de las imágenes de un estacionamiento en escala de grises.

$$\text{Precisión} = \frac{VP}{VP + FP} \quad (16)$$

Además de las métricas de efectividad y precisión, también se obtuvo el porcentaje de detección de los automóviles por imagen, esto es que, si una imagen contiene un total de 4 automóviles y el sistema logra detectar a los cuatro, entonces se dice que se alcanzó un 100% en la detección, mientras que, con respecto a la misma imagen se detectan únicamente 3 autos, entonces se dice que el sistema alcanzó un 75% en la detección; utilizando esta métrica, se logró alcanzar el 100% para imágenes con un ambiente de iluminación homogéneo, o bien, con cambios ligeros en aumento o disminución de iluminación, en cambio, cuando se presentan efectos de sombra debido a un aumento elevado de iluminación, el porcentaje de detección disminuye hasta alcanzar el 80%; esto se puede apreciar en la fig. 12, en donde las imágenes del inciso (k) y (l), presentan una baja detección de los automóviles debido a un aumento en la iluminación, además se hace una detección de falsos positivos, debido a objetos que son formados por las sombras que generan los árboles, y de igual manera, en las imágenes que se presentan en los incisos (i), (j) y (l), se

detectan falsos positivos, que corresponden a señalamientos dentro del estacionamiento y a una sucesión de la calle que se encuentra frente al mismo.

Algoritmo 1: Algoritmo propuesto para la detección de automóviles en un estacionamiento.

Input: I_{rgb}

Result: Imagen con ubicación de los automóviles I_d

- 1 Convertir I_{rgb} a espacio de color HSV $\rightarrow I_{hsv}$;
 - 2 Extraer Hue y aplicar operación de apertura $\rightarrow I_{hue}$;
 - 3 Aplicar segmentación con ICM $\rightarrow I_b$;
 - 4 Etiquetar imagen $I_b \rightarrow I_E$;
 - 5 **for** cada objeto en I_E **do**
 - 6 Obtener las coordenadas del objeto $\rightarrow [x, y]$;
 - 7 Acceder al objeto en escala de grises $\rightarrow I_{obj}$;
 - 8 Generar firma de $I_{obj} \rightarrow xi$;
 - 9 Ingresar xi a la red MLP $\rightarrow y_s$;
 - 10 **if** $y_s == 1$ **then**
 - 11 Es un auto, indicarlo en $I_{rgb} \rightarrow I_d$;
 - 12 **end**
 - 13 **end**
-

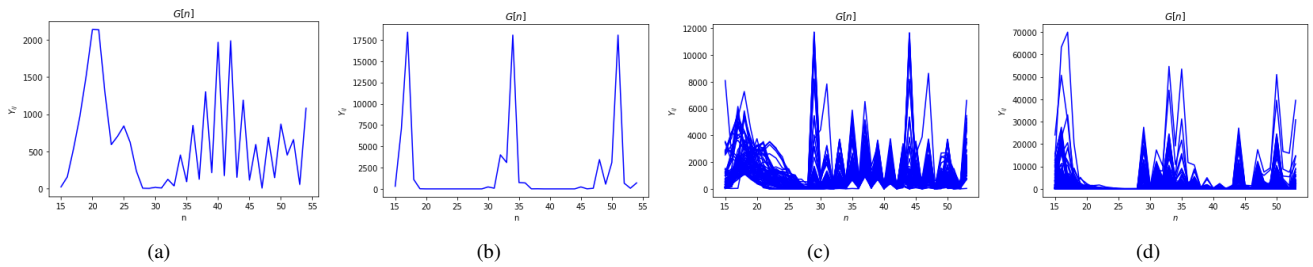


Fig. 11. Conjunto de firmas recabadas, (a) patrón de firma generado por un automóvil; (b) patrón de firma generado por un objeto distinto a un automóvil; (c) total de firmas generadas por automóviles del conjunto de imágenes de entrenamiento; (d) total de firmas recabadas por objetos que no son automóviles del conjunto de imágenes de entrenamiento.

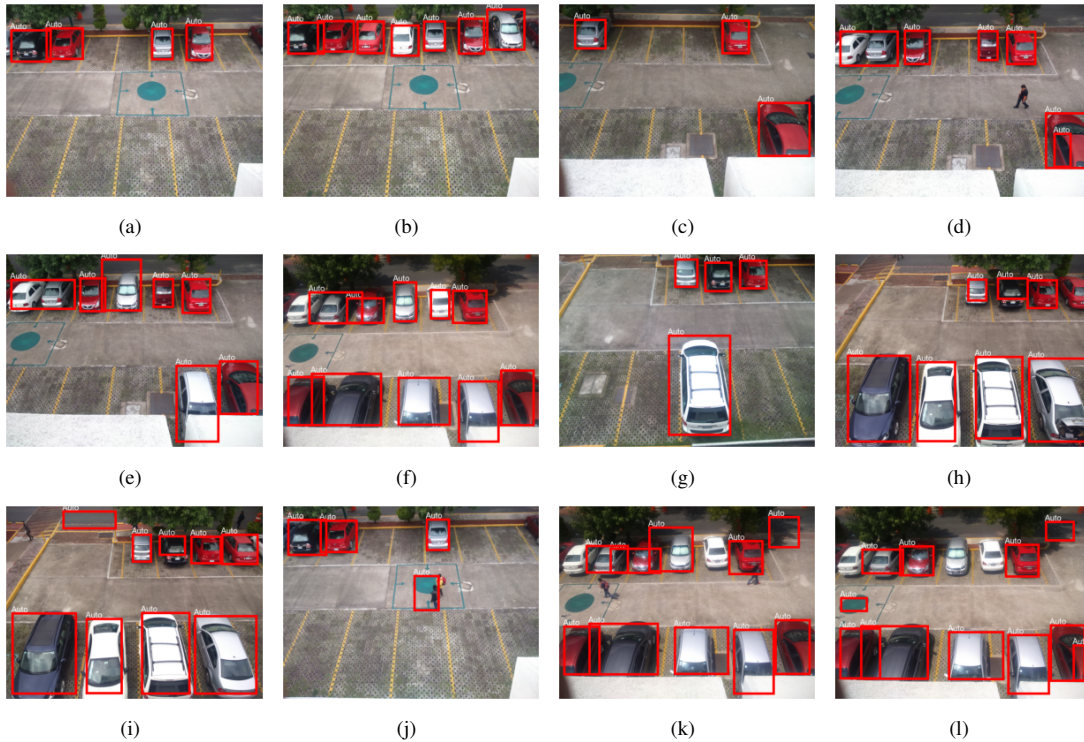


Fig. 12. Algunos resultados de la detección de automóviles obtenidos de la implementación del algoritmo 1.

Tabla II
MÉTRICAS DE LOS RESULTADOS OBTENIDOS CON LAS IMÁGENES DE PRUEBA.

Imágenes	Autos	Detectados	Efectividad	Precisión	Tiempo
86	504	483	95.83 %	92.7 %	3.16 segs.

V. CONCLUSIONES

De acuerdo con los resultados obtenidos, el método propuesto presenta un buen desempeño en imágenes donde la iluminación es homogénea o bien cuando presenta un ligero cambio, ya sea en disminución o en aumento de la iluminación, sin embargo, es susceptible a los efectos de sombra que se generan debido a una elevada iluminación ocasionada por los efectos de la luz del sol y por los árboles que se encuentran

alrededor del estacionamiento en donde se llevaron a cabo las pruebas.

Para poder solucionar algunos de estos inconvenientes, se propone para trabajos futuros combinar los métodos de extracción características por el método de la firma $G[n]$ que se obtiene con la red ICM, y características geométricas, tales como la circularidad, el perímetro, área y momentos geométricos; con los cuales se pretende eliminar los falsos positivos y mejorar así el desempeño del sistema.

REFERENCIAS

- [1] C. Huang, Y. Tai, and S. Wang, "Vacant parking space detection based on plane-based bayesian hierarchical framework", IEEE Transactions on Circuits and Systems for Video Technology, 23(9):1598– 1610, September 2013.
- [2] Soomok Lee and Seung-Woo Seo, "Available parking slot recognition based on slot context analysis", IET Intelligent Transport Systems, 10(9):594–604, 2016.

- [3] Jae Kyu Suhr and Ho Gi Jung, "Fully-automatic recognition of various parking slot markings in around view monitor (avm) image sequences", In 2012 15th International IEEE Conference on Intelligent Transportation Systems, pages 1294–1299. IEEE, 2012.
- [4] Paula Tatulea, Florina Calin, Remus Brad, Lucian Brâncoveanu, and Mircea Greavu, "An image feature-based method for parking lot occupancy", *Future Internet*, 11(8):169, 2019.
- [5] Chin-Kit Ng, Soon-Nyeon Cheong, and Yee-Loo Foo, "Lightweight deep neural network approach for parking violation detection", In Proceedings of the 2018 VII International Conference on Network, Communication and Computing, pages 332–337, 2018.
- [6] Sayani Banerjee, TS Ashwin, and Ram Mohana Reddy Guddeti, "Automated parking system in smart campus using computer vision technique", In TENCON 2019-2019 IEEE Region 10 Conference (TENCON), pages 931–935. IEEE, 2019.
- [7] Haiyan Li, Lei Guo, Pengfei Yu, Jianhua Chen and Yiyang Tang "Image segmentation based on Iterative SelfOrganizing Data clustering threshold of PCNN", 2nd International Conference on Cloud Computing and Internet of Things (CCIoT), 2016.
- [8] H. Y. Gao, X. Su, and Y. S. Liang, "Automatic Image Segmentation Using PCNN and Quantum Geese Swarm Optimization", Springer Nature, 2019
- [9] Ya Zhao and Xiaodong Gu, "Vehicle License Plate Localization and License Number Recognition Using Unit-Linking Pulse Coupled Neural Network", Springer-Verlag, 2012.
- [10] Thomas Lindbland, Jason M Kinser, and JG Taylor, "Image processing using pulse-coupled neural networks", Springer, 2013.
- [11] Yide Ma, Kun Zhan, and Zhaobin Wang, "Applications of pulse coupled neural networks", Springer, Berlin, 2010.
- [12] Ekblad U, Kinser JM, Atmer J, "The intersecting cortical model in image processing", *Nuclear instruments and methods in physics research* 525(1): 392-396.
- [13] Ivan Nunes da Silva, "Artificial neural networks", Springer, 2017.
- [14] Tom M. Mitchell "Machine learning", McGraw-Hill Science Engineering Math, March 1, 1997.
- [15] Erik Cuevas, Daniel Zarldívar y Marco Pérez, "Procesamiento digital de imágenes con MATLAB & Simulink", Ra-Ma, 2013.
- [16] Genesan P, Rajjini V, Satish B. S. y Shaik K. B., "HSV color space based segmentation of region of interest in satellite images", IEEE international conference on control, Instrumentation, Communication and Computational Technologies, 2014, 101-105.
- [17] Nobuyuki Otsu, "A threshold selection method from gray-level histograms", 9:62-66, 1979.
- [18] Xiaodong Gu, "Feature extraction using unit-linking pulse coupled neural network and its applications", Springer science, 2007.
- [19] Zhaobin Wang, Xiaoguang Sun, Yaonan Zhang and Zhu Ying, "Leaf recognition based on PCNN", *Natural computing applications forum*, 2015.
- [20] A. Samir Elons and Magdy Aboull-Ela, "Arabic sign language recognition system based on adaptive pulse-coupled neural network", AMLTA, 2012.
- [21] Stephen Marsland, "Machine learning an algorithm perspective", CRC Press, 2015.

License plate detection based on Intersecting Cortical Model and entropy signatures

Elizabeth Xicotencatl-Flores

Benemérita Universidad

Autónoma de Puebla

Puebla, México

elizabeth.xicotencatlf@alumno.buap.mx

Aldrin Barreto-Flores

Benemérita Universidad

Autónoma de Puebla

Puebla, México

aldrin.barreto@correo.buap.mx

Salvador E. Ayala-Raggi

Benemérita Universidad

Autónoma de Puebla

Puebla, México

salvador.raggi@correo.buap.mx

José F. Portillo-Robledo

Benemérita Universidad

Autónoma de Puebla

Puebla, México

francisco.portillo@correo.buap.mx

Verónica E. Bautista-López

Benemérita Universidad

Autónoma de Puebla

Puebla, México

veronica.bautistalo@correo.buap.mx

Abstract—In recent years, ALPR systems has been a growing research area due to the multiple applications involved. Many approaches were proposed in literature but there are still elements to optimize that might increase the recognition ratio, for example, license plate detection. In this work, we introduce an algorithm based on ICM (Intersecting Cortical Model), a simplified pulse coupled neural network in which no training is needed. It is used for license plate location and K nearest neighbor (K-NN) for image classification. After analyzing potential license plate candidates, a single region is selected based on feature extraction and the signature created by the entropy of the ICM pulses. The K-NN process these signatures by evaluating the distance between them to detect the image class. It is shown that the proposed methodology works efficiently on license plates that have similar format of some Mexican states (Puebla, Morelos, Tlaxcala, Tabasco, Hidalgo, Quintana Roo, Guanajuato and Mexico City). It achieved an accuracy of 92.4%.

Index Terms—License plate identification, Intersecting Cortical Model (ICM), K nearest neighbor (K-NN).

I. INTRODUCTION

Automatic license plate recognition (ALPR) helps in collecting vehicle information reducing the manual work. They have been used in many intelligent transportation applications, from detecting spaces in a parking lot [1] and access control [2] to vehicle monitoring [3]. It has been studied worldwide by several researchers and still is an open problem due to the large variability in image acquisition conditions (non-uniform illumination during the day, occlusion, distance ranges between the camera and the vehicle, blurring, number of objects in the scene). Also the plate format assigned by each country is different and is further complicated by the fact that it may need to operate in real time.

ALPR consists of three main stages: license plate detection (LPD), character segmentation and recognition. The first stage is the key which directly affects the following recognition modules and this article will focus only in this step. The second and third stage is mostly done by template matching and optical character recognition but can be achieved by different

techniques. This paper is organized as follows: Section II gives an overview of plate detection and related work. Section III presents a detail description of the proposed algorithm. Section IV shows the experimental results and the conclusion is given in Section V.

II. RELATED WORK

The existing algorithms for license plate detection are based on machine / deep learning and image processing techniques that work for example with object detection, edge, color and texture [4]. It is inappropriate to declare which method has the highest performance since there is a lack of uniform way to evaluate them. However, they can be combined to increase their accuracy or be adapted to new needs.

In [5], Safaa S. and Jumana A. have used the size of the plate and morphological operations. However, its performance is limited in complex scenarios. Some studies have used Sobel filter [6-8] combined with horizontal and vertical projections but as a limitation, it is sensitive to unwanted edges and not suitable to be applied with blurry images or where the number of edges is greater in the background than in the area of the plate. In the last years, the trend is towards multiple plate recognition and the use of robust neural networks architectures for complex environments such as ALPRNet [9], optimal K-means with convolutional neural network [10], multi-task CNN [11], YOLOv3 [12], VGG, RCNN and ResNet [13]. On this last case, the computational cost must be considered and is dependent on the hardware platform chosen.

III. METHODOLOGY

As mentioned, several factors make ALPR systems a challenging including finding a Mexican plate dataset to train a custom algorithm. For this research different captures of license plates were obtained under various illumination scenarios considering the rear and front view of the vehicles captured at a distance between 50cm and 1.5m using a smartphone

with a camera resolution of 13 megapixels. The plate's format includes a serie of three letters followed by four numbers, a design used in some Mexican states to identify the origin of vehicles.

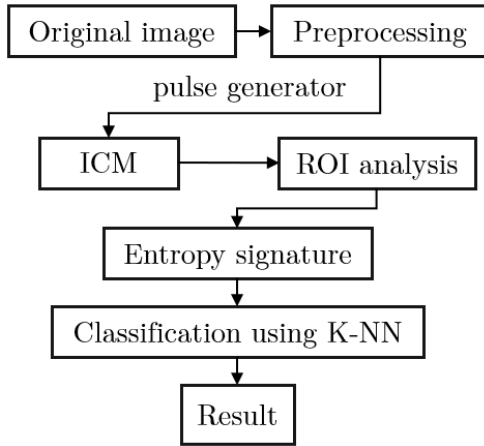


Fig. 1. Block diagram of proposed method

The steps of the proposed LPD are summarized in Fig.1 and they will be explained below.

A. Preprocessing

Image preprocessing is necessary for segmentation. First, the input color image is transformed to grayscale considering the average of the three RGB channels. The contrast is improved with an adaptive histogram equalization to make a better difference between the vehicle and the plate. Histogram equalization is an image processing technique for adjusting the image's intensity that aims a uniform distribution of all values that tends to increase the noise. With adaptive histogram equalization, the image is divided into an $M \times N$ grid and the equalization is applied as usual to each cell which implies that the contrast enhancement is local rather than global. The histogram equalization formula used on a 20×20 kernel was:

$$f_{ac} = (p - p_{low}) \left(\frac{p_{max} - p_{min}}{p_{high} - p_{low}} \right) \quad (1)$$

where the pixel value p is linearly interpolated from the interval $[p_{low}, p_{high}]$ to $[p_{min}, p_{max}]$.

After that, unwanted information is removed to increase the image quality using a gaussian filter with a 5×5 kernel (shown in Fig.2) that represents a discrete approximation of a Gaussian distribution. Unlike other smoothing filters, it does not remove the edges of the image. Finally, the image pixels are scaled from uint8 type $[0, 255]$ to float32 values within the interval $[0, 1]$.

B. Segmentation

The intersecting cortical model (ICM) facilitates the umbralization process giving multiple binary images (pulses) $Y[N]$ depending on a fixed number of iterations and can also be used as a recognition method [14].

$$\frac{1}{57}$$

0	1	2	1	0
1	3	5	3	1
2	5	9	5	2
1	3	5	3	1
0	1	2	1	0

Fig. 2. Gaussian kernel

The ICM is a simplified version of a pulse coupled neural network (PCNN) that removes the *Linking* neurons of the original model reducing the adjusting parameters. It was designed to minimize computational cost in image processing and is described by:

$$F_{ij}[n+1] = fF_{ij}[n] + S_{ij} + W\{Y[n]\}_{ij}, \quad (2)$$

$$Y_{ij}[n+1] = \begin{cases} 1 & \text{if } F_{ij}[n+1] > \Theta_{ij}[n] \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$\Theta_{ij}[n+1] = g\Theta_{ij}[n] + hY_{ij}[n+1] \quad (4)$$

In this equations, the *Feeding* channel F_{ij} depends on the current values of each pixel, their neighbours and the feedback from the network. S_{ij} is the normalized grayscale image (the values are in the range of 0-1), Θ_{ij} is a dynamic threshold, W_{ij} is the connection function which calculates the communication from the neighboring neurons that can be a 5×5 or 3×3 matrix [15] and Y_{ij} is the output image. This process is represented in Fig.3.

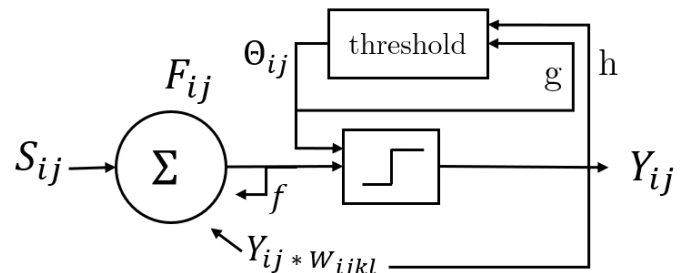


Fig. 3. Intersecting Cortical Model

f , g , and h are constants that fit the model, forcing the ICM to pulse in a synchronous form to perform grouping of uniform regions. In this network, no training procedure is needed and the main idea is to force the ICM to generate smaller regions than the ones generated in the first iteration, adjusting the parameters to make a dynamic process.

Shannon entropy was used as a metric for the segmentation process. It is a property that reflects the amount of information contained in images and can be computed by:

$$H(s) = -p_1 \ln p_1 - p_0 \ln p_0 \quad (5)$$

where p_1 and p_0 are the probabilities of 1 and 0 in the segmented image respectively. Fig.4 shows an example of the

segmentation process using the Shannon's maximum entropy to select the suitable pulse over twenty iterations.



Fig. 4. Segmentation process with the highest entropy.

C. Region of interest analysis

After selecting the image with the highest entropy, erosion operator is applied using a 6×6 kernel initialized with ones. Erosion (represented by the symbol \ominus [16]) is a morphological process designed for binary images and extended to grayscale. The kernel scans the image, and finds the overlapping interval between the kernel and the image pixel. The output is the minimum value of all pixels in the neighborhood. In a binary image, a pixel is set to 0 if any of the neighboring pixels have the value 0. The basic effect of the operator is to erode away the boundaries of regions of foreground pixels, eliminating small details acting as a local minimum filter.

Then, the image is divided into smaller parts to analyse the shape of the extracted regions. Only the rectangle boundaries are considered as objects. Those values are found by evaluating every image coordinate (x,y) to find the minimum and maximum to obtain the new coordinates $(X=\min(x), Y=\min(y))$ and dimensions $(width = (\max(x)-\min(x))$ and $height = (\max(y)-\min(y))$ of every rectangle. During the loop, if the current value is lower or higher than the previous min or max x,y , then, the above will be replaced.

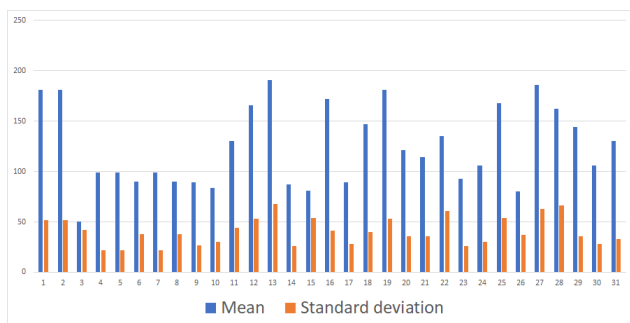


Fig. 5. Mean and standard deviation of 31 license plates

The resulting rectangles had different areas related to the initial conditions of the images on the dataset (according to the object camera distance) but it was possible to define a specific area ($width \times height = [3000 ; 60000]$) and aspect ratio ($[1.4 ; 3]$) that corresponds to the size for potential license plate candidates:

$$Aspect\ ratio = width/height \quad (6)$$

The final choice is based on the maximum arithmetic mean of the grayscale values of those regions to analyze the intensity distribution of all plates obtained from the dataset. Fig.5 shows the mean and standard deviation of thirty one license plates and can be concluded that they have a low contrast.

In this step, it was also considered to use the skeleton but in some regions the number of edges is greater than the ones on the license plate. This issue is attributed to stickers and ornaments on the vehicle.

D. Signature

Time series in PCNN was first introduced by Johnson [17] as an image feature extraction method suitable for statistical classification and is defined as the sum of the neurons that fire at each iteration, being a vector that can be described by the following equation:

$$G[n] = \sum_{ij} Y_{ij}[n] \quad (7)$$

The output $Y_{ij}[n]$ of ICM is the pulse sequence, and the output of the binary image entropy is a time series varying with the number of iterations. It was used to reduce the image dimensions (the amount of computation and storage of data is reduced) and was considered in view of the fact that it is invariant to geometric distortions (scaling, rotation and translation). The series of pulse images can also be converted into entropy signatures by:

$$En[n] = -P_1 \log_2(P_1) - P_0 \log_2(P_0) \quad (8)$$

where P_1 is the probability of 1's in a binary image, and P_0 is of 0's in the binary image. The example in Fig.6 shows that entropy sequence is independent of rotation.

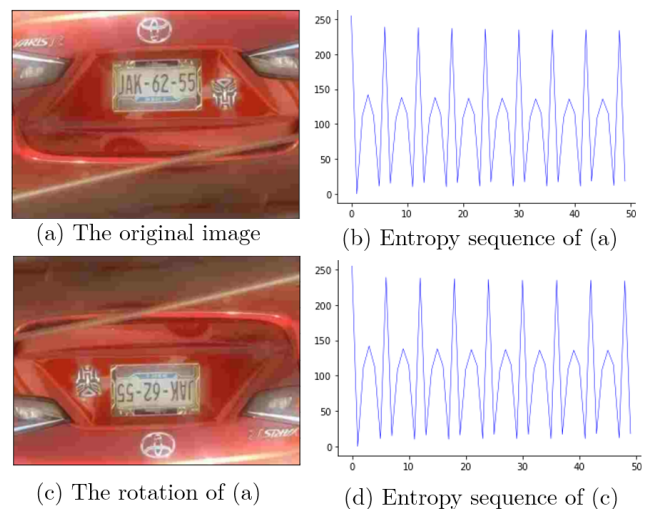


Fig. 6. Entropy signatures

E. K-NN Classifier

To verify whether the feature parameters that were obtained are the right ones, it was used k nearest neighbors (K-NN). It is a supervised machine learning algorithm that can be used for both classification as well as regression. It does not make assumptions about the distribution of the data and the classification is based on the search for similar characteristics with each of the elements that are available in the training set. Predictions are obtained by using a distance function to calculate the proximity of all training points to the new data to be labeled and then choosing the mode. The value of k is the number of closest neighbors to perform the comparison and carry out the voting. The algorithm can be summarized as:

- 1) Load the training and testing dataset
- 2) Choose the k value
- 3) Calculate the distance between image test and each training data. The most used is the Euclidean.

$$D(a, b) = \sqrt{\sum_{i=1}^n (b_i - a_i)^2} \quad (9)$$

- 4) Sort the distance in ascending order
- 5) Choose the top k rows from the sorted list
- 6) Assign a class based on most frequent class

IV. EXPERIMENTAL RESULTS

In this section, the performance of the ICM signature was evaluated on 411 images. They present variations in lighting conditions and some of them blurring, due to the remoteness of the capture. Full plates are shown in all photos. The original dimensions were changed to 400x550 for further processing. In Fig.7 are some license plates from the dataset.



Fig. 7. Some samples from the dataset

For classification, it was used cross validation where the total of samples were divided into a set of ten blocks, nine for training and the remaining for test. The recognition rate was obtained by dividing the number of samples that were classified correctly by the total of testing samples multiplied by 100. The performance was evaluated per block and then averaged contemplating thirteen neighbours.

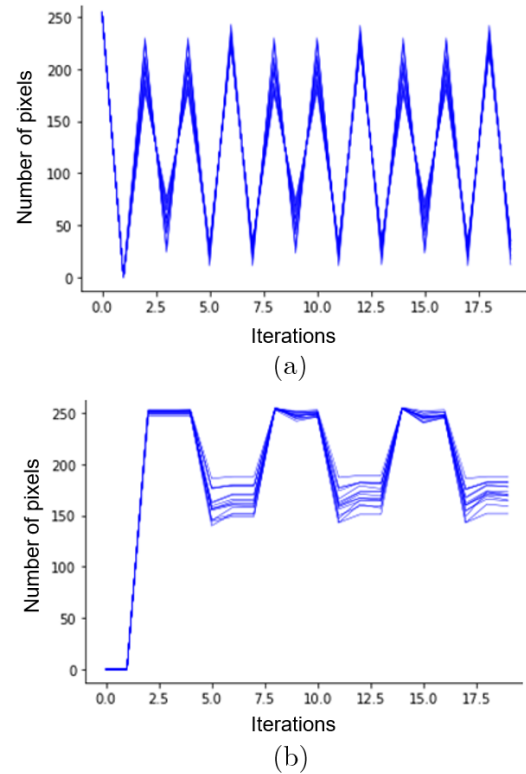


Fig. 8. (a) ICM time signature and (b) entropy sequence of 15 license plates

To illustrate the effectiveness of the proposed algorithm Fig.8 shows the (a) time ICM series and the (b) entropy sequence between fifteen license plates, where horizontal axis denotes the number of iterations. It can be noticed that they are very similar and that image feature dimensions are greatly reduced. The pattern repeats the bandwidth and number of waves of each image, although the amplitude is variable. Both results were normalized using a min-max normalization and implemented on the process.

The recognition ratio for all the samples on the dataset using the proposed method is 88.8% with time signatures and 92.4% with entropy sequence. ICM parameters were obtained empirically considering the network behavior, being $f = 0.2$, $g = 0.01$ and $h = 75$ the values that denoted better results.

V. CONCLUSIONS

Nowadays exists many algorithms for license plate location but each one has its own restrictions. As an example, there are many methods of image segmentation, the most recognized is using a threshold but is very difficult to select an appropriate

value for all the data on a training set. A methodology of license plate detection based on ICM has been presented in this paper. As it is shown in the experimental results, license plates features can be extracted based on their unique invariant entropy signatures and then, calculate their Euclidean distance to determine the final recognition results. The recognition rate proves the good performance of the proposed method. It is important to highlight that the number of iterations is dependant on the cyclical response of the ICM network.

REFERENCES

- [1] Chandra, Hans, et al. "Smart parking management system: An integration of RFID, ALPR, and WSN." 2017 IEEE 3rd International Conference on Engineering Technologies and Social Sciences (ICETSS). IEEE, 2017.
- [2] Mohandes, M., et al. "An intelligent system for vehicle access control using RFID and ALPR technologies." *Arabian Journal for Science and Engineering* 41.9 (2016): 3521-3530.
- [3] Amonkar, Saidatt, Anikumar Naik, and Amogh A. Sanzgiri. "ALPR System Using Sequence Modelling: A real time system for vehicle authentication." 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE, 2019.
- [4] Shashirangana, Jithmi, et al. "Automated License Plate Recognition: A Survey on Methods and Techniques." *IEEE Access* (2020).
- [5] Omran, Safaa S., and Jumana A. Jarallah. "Iraqi License Plate Localization and Recognition System Using Neural Network." 2017 Second Al-Sadiq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA). IEEE, 2017.
- [6] Lin, Haibo, et al. "License plate location method based on edge detection and mathematical morphology." 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). Vol. 1. IEEE, 2020.
- [7] Xiong, Chunrong, and Wenming Huang. "License plate location based on compound mathematical morphology." 2009 Third International Conference on Genetic and Evolutionary Computing. IEEE, 2009.
- [8] Hongping, Hu, and Bai Yanping. "A kind of license plate location based on mathematical morphology and edge detection." *Proceedings of 2011 International Conference on Electronic & Mechanical Engineering and Information Technology*. Vol. 5. IEEE, 2011.
- [9] Huang, Qiuying, Zhanchuan Cai, and Ting Lan. "A Single Neural Network for Mixed Style License Plate Detection and Recognition." *IEEE Access* 9 (2021): 21777-21785.
- [10] Pustokhina, Irina Valeryevna, et al. "Automatic vehicle license plate recognition using optimal K-means with convolutional neural network for intelligent transportation systems." *IEEE Access* 8 (2020): 92907-92917.
- [11] Wang, Wanwei, et al. "A light CNN for end-to-end car license plates detection and recognition." *IEEE Access* 7 (2019): 173875-173883.
- [12] Henry, Chris, Sung Yoon Ahn, and Sang-Woong Lee. "Multinational license plate recognition using generalized character sequence detection." *IEEE Access* 8 (2020): 35185-35199.
- [13] He, Ming-Xiang, and Peng Hao. "Robust Automatic Recognition of Chinese License Plates in Natural Scenes." *IEEE Access* 8 (2020): 173804-173814.
- [14] Wang, Zhaobin, et al. "Leaf recognition based on PCNN." *Neural Computing and Applications* 27.4 (2016): 899-908.
- [15] Ortiz Rangel, Estela, Manuel Mejía-Lavalle, and Humberto Sossa. "Filtrado de ruido Gaussiano mediante redes neuronales pulso-acopladas." *Computación y Sistemas* 21.2 (2017): 381-395.
- [16] Montes, Javier Rojas. "Procesamiento digital de imágenes con MatLAB y SIMULINK." *Sistemas Telemática* 10.21 (2012): 77-79.
- [17] Lindblad, Thomas, et al. *Image processing using pulse-coupled neural networks*. Heidelberg: Springer, 2005.